EXHIBIT 2

IN THE UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF CALIFORNIA
SAN FRANCISCO DIVISION

ORACLE AMERICA, INC.,	)	
	)	
Plaintiff,	)	
	)	
v.	)	Civ. A. No. 10-03561 WHA
	)	
GOOGLE INC.,	)	*(Jury)*
	)	
Defendant.	)	

EXPERT REPORT OF PROFESSOR DOUGLAS C. SCHMIDT, Ph.D.

January 8, 2016

Highly Confidential – Attorneys' Eyes Only

Table of Contents

I. ASSIGNMENT

1. I have been asked to provide my analysis and opinions concerning: (1) the general history of the Java platform, including the development of its API packages, (2) the compatibility of Google's Android platform with the Java platform, e.g., the compatibility of programs written for the Android platform with the Java platform, and (3) the importance of the Java API packages to the Android platform, specifically whether the Android platform will operate without the Java API packages. In particular, I have been asked to objectively consider whether Java programs can be executed correctly in the environs of the Android platform, and how the results indicated in these inquiries have impacted developers, end-users, Sun Microsystems, Oracle America, and the Java community at large.

2. In addressing these questions, I draw on my academic and professional background, which includes more than 30 years of experience in the areas of computer programming, software engineering, digital architecture, technology adoption and diffusion, the creation and management of networked systems, object-oriented middleware, mobile cloud computing infrastructure and applications, and the marketplace dynamics of software and information systems.

3. I am being compensated for my work on this case at a rate of $550 per hour. My compensation is not contingent upon my testimony or on the result of this proceeding.

4. My work is ongoing, and I reserve the right to modify or supplement my conclusions as additional information becomes available to me, or as I perform further analysis.

II. QUALIFICATIONS

5. I am currently a tenured Full Professor with the Department of Electrical Engineering and Computer Science at Vanderbilt University in Nashville, TN, where I also serve as the Associate Chair of the Computer Science and Engineering program. I have been a full-time university professor since 1994, and I was previously a tenured professor at both the University of California, Irvine and Washington University in St. Louis, MO.

6. I hold a Doctor of Philosophy (PhD) degree in Computer Science from the University of California (UC), Irvine in Irvine, CA. I also hold a Master's Degree in Computer Science from UC

Irvine, as well as a Bachelor's Degree and Master's Degree in Sociology from the College of William and Mary in Williamsburg, VA. Prior to completing my graduate studies at UC Irvine I worked with the Urban Information Systems ("URBIS") project at UC Irvine's Public Policy Research organization, where I studied end-user computing interactivity with municipal institutions in 40 cities across the United States. I also worked at the International Center for Information Technology ("ICIT") in Washington D.C. on projects assessing techniques for improving software productivity.

7. During the past 30 years, I have conducted and supervised a significant number of research projects involving a wide range of software-related topics including patterns, optimization techniques, and empirical analyses of object-oriented middleware frameworks for distributed real-time embedded systems and mobile cloud computing applications. I have also supervised the research of more than 40 PhD and Masters graduate students to date.

8. From 2010 to 2014 I served as a member of the United States Air Force Scientific Advisory Board, where I was the Vice Chair of Cyber Situational Awareness, a study for the U.S. Air Force on the network security of mission operations. I also serve on the Advisory Board for the U.S. Naval Air Systems Command (NavAir) Future Airborne Capability Environment (FACE) and was recently a co-lead of a task force on "Published Open Interfaces and Standards" for the U.S. Navy's Open Systems Architecture initiative.

9. From 2000 to 2003 I served as a Deputy Office Director and Program Manager at the Defense Advanced Research Projects Agency (DARPA), where I led the national research and development effort on portable open system architecture middleware for distributed real-time and embedded (DRE) systems. During that time I also co-chaired the Software Design and Productivity (SDP) Coordinating Group of the U.S. government's multi-agency Networking and Information Technology Research and Development (NITRD) Program, which helped to formulate a national interagency software research agenda.

10. I have been programming with object-oriented languages since the mid-1980s, when I began to program with C++. I have been programming with Java since the mid-1990s. While at Washington University in St. Louis I led the development of one of the first Java frameworks for concurrent and networked middleware and applications (Java ACE). Since 2012, I have taught

Java and Android to more than 300 students in face-to-face courses at both the undergraduate and graduate levels at Vanderbilt University. Since 2013, I have taught Java and Android to more than 200,000 students in Massive Open Online Courses (MOOCs) on the Coursera platform.

11. In addition to my regular course offerings, I have also taught hundreds of short-courses and tutorials on numerous subjects, including: object-oriented design patterns and programming techniques; systems programming and network programming for UNIX and Windows; the C++, C, and Java programming languages; and, various courses on distributed operating systems, compiler construction, algorithms, and data structures.

12. For the past three decades, I have led the development of ACE, Java ACE, TAO, and CIAO.[1] The millions of lines of C++ and Java code in these frameworks provide layers of system architecture middleware that simplify the development of concurrent and networked software.

13. I have published more than 570 scholarly articles and technical papers, and I am the co-author or editor of 12 books on various aspects of software architecture, network programming, object-oriented frameworks, distributed and real-time systems, open-source middleware platforms, and mobile cloud computing applications. My work has been cited more than 25,000 times across a comprehensive spectrum of high-impact publications, and my current h-index score is 74, which indicates the significant impact of my publications on scholarly literature in the field of Computer Science. [2]

14. I have served on the editorial board of numerous journals, including publications by IEEE and the ACM, and I have been a guest editor of numerous special issue journals based on my research expertise. Most recently, I served as a guest editor of the Springer *Annals of*

[1] *See* Obtaining ACE, TAO, CIAO, and DAnCE, Vanderbilt, http://download.dre.vanderbilt.edu/ (for access to ACE, TAO and CIAO downloads). *See* Java ACE, Vanderbilt, http://www.dre.vanderbilt.edu/JACE/ (for access to Java ACE downloads).

[2] The h-index is a popular measure of scholarly productivity. The definition of the index is that a scholar with an index of h has published h papers each of which has been cited in other papers at least h times. Thus, the h-index reflects both the number of publications and the number of citations per publication.

Telecommunications special issue "Middleware for Internet Distribution in the Context of Cloud Computing and the Internet of Things."[3]

15. My research has been funded by a variety of organizations, including both federal agencies, such as DARPA, NSF, NASA, NIH, the U.S. Air Force, and the U.S. Navy, as well as leading companies, such as Northrup Grumman, Raytheon, Lockheed-Martin, Boeing, McDonnell-Douglas, General Electric, and Siemens.

16. I have received a number of other honors and awards, including election to professional organizations, engagements for invited talks and, most recently, the 2015 Award for Excellence in Teaching from the Vanderbilt University Department of Electrical Engineering.

17. Over the last two decades I have been retained as an expert consultant more than two dozen times in a variety of computer software-related matters, focusing primarily on topics related to the software and network infrastructure of mobile devices.

18. My complete CV is in Appendix A.

III. MATERIALS RELIED UPON

19. In drafting this report, I have relied upon a number of different public sources, official declarations, and expert reports from the present matter, original empirical analyses, and my own background, experience and knowledge of certain topics and events relevant to the issues at hand in this case.

20. Public sources I have relied on include: source codes from the Android and Java platforms; documents, press releases, user guides, and other statements published on the websites of Oracle, Google and other technology companies; various indices of computer programming activities; news, developer forum and journal publications; scholarly literature and white papers; and documents from the first trial between the parties and the appeal.

[3] Springer/Annals of Telecommunications 2014: Special Issue on Middleware for Internet distribution in the context of cloud computing and the Internet of Things, WikiCFP, http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=38110 (last visited January 7, 2016).

21. Official declarations from the present matter I relied upon include: the declaration of Dr. Mark Reinhold concerning compatibility testing and the expert report of Mr. Robert Zeidman concerning Google's copying of Oracle's code. My understanding from my review of Mr. Zeidman's report is that Google has copied the declaring code and the structure, sequence and organization of 37 Java API packages – I refer to these copied elements throughout this report as "the 37 Java APIs".

22. Original empirical analyses I relied upon include: evaluations, measurements, assessments, and calculations related to programming environment attributes; the timing and composition of Java and Android version releases; the interpretation of compatibility test results provided to me by Dr. Reinhold; the interpretation of copying analysis provided to me by Mr. Zeidman; the performance of various Android build testing; and the performance of various Android and Java runtime and machine testing.

23. I have also relied upon my own experience and background as a professor, researcher, published author, and lifelong student of computer science topics. In particular, I have drawn on my specific knowledge concerning programming in Java and Android environments, as I have taught courses on both platforms to thousands of people during the past decade.

24. A list of the specific materials that I have relied upon and considered in preparing and drafting this report is available in Appendix B – Materials Considered.

IV. SUMMARY OF OPINIONS

25. Since its inception in the mid-1990s, the Java platform has had significant impact on software programming and has fostered a robust, dynamic community of application and infrastructure developers. Java API packages offer the cogent, portable set of materials that enable developers to more effectively create programs using the Java platform. Compatibility is an essential virtue of the Java platform and the Java API packages in particular, as indicated by the "Write Once, Run Anywhere" philosophy put forth by Sun Microsystems to emphasize Java's

cross-platform benefits with respect to portability and underlying technology architecture.[4] Both Sun and Oracle have actively sought to preserve compatibility.[5]

26. Google's copying of the 37 Java APIs is a reflection of the value of the Java API packages' key attributes, including benefits to developer productivity (e.g., leveraging the large pool of developers familiar with the Java API packages) and benefits related to the creative expression reflected in the packages, as well as avoiding the technical downsides of alternative platforms available during the mid-2000s.

27. Google's use of the 37 Java APIs in the Android platform violates the fundamental "Write Once, Run Anywhere" philosophy of the Java ecosystem. In addition to failing to meet the expectations of compatibility that Oracle has fostered in the community, Google's use of the 37 Java APIs in Android represents both a subset and superset of the Java SE API Specification and thus fails to demonstrate compatibility with Java when subjected to various forms of compatibility testing, including Oracle's industry benchmark Technology Compatibility Kit ("TCK," or sometimes referred to as or used interchangeably with the Java Compatibility Kit or "JCK"), as well as independent tests and measurements. Moreover, Google's incompatible[6] use of the 37 Java APIs in Android can be seen in both Android's software development kits ("SDKs") and Android's virtual machine runtime environments.

28. The Android platform is critically dependent on the 37 Java API packages at issue, individually and collectively, and the copied declaring code. I demonstrate this critical dependence by showing that without the 37 Java API packages at issue, individually and collectively, and the copied declaring code, the build process used to create the code that runs Android on mobile devices fails to generate the necessary executable code (called executable

[4] Lindholm, T., Yellin, F., Bracha, G., & Buckley, A. (2014). *The Java Virtual Machine Specification*. Pearson Education., p. 1 ("The java platform was initially developed to address the problems of building software for networked consumer devices. It was designed to support multiple host architectures and to allow secure delivery of software components. To meet these requirements, compiled code had to survive transport across networks, operate on any client, and assure the client that it was safe to run.") http://tech-insider.org/java/research/1996/0123.html

[5] The compatibility guide for Java 8 is described at http://www.oracle.com/technetwork/java/javase/8-compatibility-guide-2156366.html.

[6] When I describe "incompatible" uses in this report, unless otherwise noted, I mean the incompatibility between Google's use of the 37 API packages and the suite of API packages available in various editions of Java SE (JDK). When I write about "compatibility" in this report, unless otherwise noted, I mean conformance with the Java Compatibility Kit (JCK).

image files). The build process fails if files from even one of the 37 Java API packages at issue is removed. In fact, the build process also fails when I remove only the copied lines of declaring code while leaving the rest of the files in place. As a result, Android will not compile, let alone run on a mobile device without all of the copied declaring code and the full set of files for the 37 Java API packages. In short, without any or all of 37 Java API packages at issues and the copied declaring code, mobile devices running Google's Android are unusable.

V. BACKGROUND ON THE JAVA AND ANDROID PLATFORMS

A. Background on the Java Platform

29. Sun Microsystems first released the Java programming platform in 1995. In 2010, Oracle Corporation (also "Oracle") acquired Sun. The Java platform contains three distinct resources: the Java Language, the Java Virtual Machine, and the Java API packages (sometimes referred to as the Java Class Libraries).[7] Java is one of the most popular programming platforms in the world.[8] The Java platform, including the Java API packages at issue in this case, evolved through multiple iterations, as is shown in Appendix D – Evolution of Java APIs. The Java API packages enable and accelerate the development of a wide range of different programs and applications across a wide range of operating systems and hardware platforms. In this section, I provide a description of the Java Platform and Java API packages. I provide a glossary of terms relevant to this section in Appendix E - Glossary of Terms.

30. The Java platform, and particularly the Java API packages, enables a large and active community of software developers. Those software developers create computer programs (also known as applications or simply apps). The programs run on computing devices like personal computers and mobile phones. The group of software developers who write programs for a given platform are known as that platform's community (e.g., the Java community).

[7] A representation of the Java platform is at Appendix C - The Java Class Libraries. *See* JDK 5.0 Documentation, Sun, https://web.archive.org/web/20100330080522/http://java.sun.com/j2se/1.5.0/docs/index.html (last visited January 7, 2016).

[8] Stephen Cass, *The 2015 Top Ten Programming Languages,* IEEE Spectrum (July 20, 2015) http://spectrum.ieee.org/computing/software/the-2015-top-ten-programming-languages.

31. The Java platform and the API packages in particular were designed to make programs portable - as the Java philosophy "Write Once, Run Anywhere" emphasizes.[9] Portability enables programmers to write programs for one platform and have their programs run similarly on any other combination of hardware and operating system that contained the necessary Java platform resources. For end devices to run these programs, they must have the Java Runtime platform resources.

1) The Java Programming Language

32. Programming languages enable software developers (also known as programmers) to write lines of source code. Source code is written by and/or read by human developers. Source code with consistent syntax is ultimately converted into instructions that direct computing devices to perform specific actions. Typically a developer will write a program that contains multiple lines of source code, and those lines of source code will be converted into machine instructions (known as "compiling") that tell the computer running the program what to do.

33. The Java Language is object-oriented.[10] Object-oriented languages bundle methods, behaviors and data into "objects" that can interact with one another. Object-oriented programming is a popular approach to software development,[11] and is the basis of widely used programming languages, such as Java, C++, C#, Python and Ruby.

34. Programmers can write source code that creates objects and instructs them to interact with one another. In object-oriented languages, source code defines classes with methods and fields. A class defines the properties and behaviors of objects created by that class (i.e., a class is a blueprint from which object instances are created). Methods are members of a class that

[9] "Java was based on the idea that the same software could run on many different kinds of computers, consumer gadgets, and other devices on a network. The big promise of Java was that it would allow applications to run on different types of computers without having to be rewritten. Java applications would run on 'virtual machines" that were written for each platform."; "By running on a software layer above the operating system, Java would make applications portable across many platforms. This promise of 'Write Once, Run Anywhere' had enormous value for software developers who previously had to write different versions of the same application to run on various operating systems (Unix, Windows, Macintosh, etc.)." J2EE vs. Microsoft .NET, IBM, http://www-01.ibm.com/software/smb/na/J2EE_vs_NET_History_and_Comparison.pdf (last visited January 7, 2016).

[10] This document describes the Java object-oriented language: https://docs.oracle.com/javase/specs/jls/se8/jls8.pdf.

[11] This article Stephen Cass, *The 2015 Top Ten Programming Languages,* IEEE Spectrum (July 20, 2015) http://spectrum.ieee.org/computing/software/the-2015-top-ten-programming-languages describes the top ten most popular programming languages in 2015, many of which are object-oriented.

describe each behavior for an object. Fields specify variables that are used to capture the state of an object by storing values.

35. Programmers can also create interfaces that are like classes, but do not themselves contain implementations of methods. Object-oriented languages, including Java, further enable developers to define relationships between classes and between interfaces through inheritance, which allows developers to cause one class to acquire the properties (methods, fields, etc.) of another class. For example, through inheritance two classes may be defined in a relationship by which one class extends another class.

36. The Java programming language provides the syntax for developers to write source code defining classes and methods that are subsequently converted to machine code and executed on computing devices. For example, the Java language syntax indicates how to formulate "if" statements that direct the flow of control through a program, making conditional choices in that program (e.g., if it is 9 a.m., have coffee; if it is noon, have lunch). Another example of syntax is "for" loops that can run a certain block of code multiple times.

2) The Java Virtual Machine

37. Once a programmer has written lines of source code that constitute a program in the Java programming language, the program is compiled into an intermediate form of instructions called bytecode, which can be executed on a computing device. The program is executed in an execution environment called the Java Runtime Environment ("JRE"). When the JRE is present on a computing device then a Java program can executed on that computing device. The JRE includes the Java Virtual Machine ("JVM") and the Java Class Libraries (i.e., the API packages). The Java Virtual Machine[12] is responsible for executing the bytecode instructions of the program together with those of the class libraries.[13]

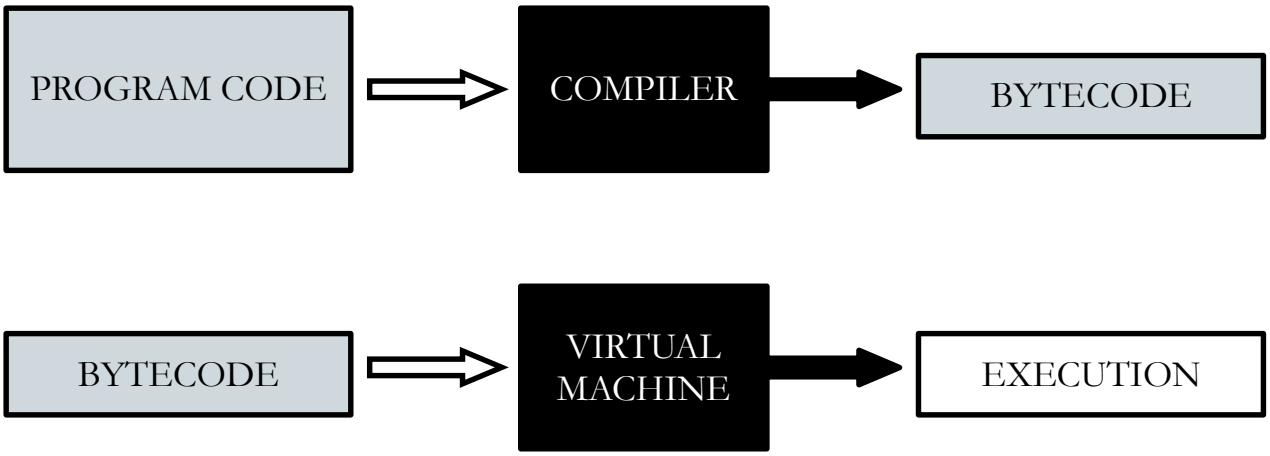
38. In other programming languages, such as C or C++, it is necessary to use a compiler to compile the program source code into native machine-language code that is specific to the architecture of the underlying computer hardware. In contrast, by compiling the Java language

[12] Tim Lindholm & Frank Yellin, The Java Virtual Machine Specification (Addison-Wesley Longman Publishing Co., Inc, 2d ed. 1999).

[13] Colloquially the term "JVM" is often used when what is really meant is the full Java Runtime Environment ("JRE").

code into the intermediate bytecode form, the Java program can be executed by a running instance of the JVM. The presence of a JVM instance on a computing device enables the same Java bytecode program to run on a variety of different types of computing devices. In Figure 1 below, I depict this execution process (Virtual Machine represents the JVM).

Figure 1: The Two-Step Process for running a Java program



39. Unlike machine language code, Java bytecode is processed by an instance of the JVM, rather than being processed directly by the "real" computer hardware processor. The JVM can either interpret the bytecode directly, step-by-step, or else it can convert the bytecode into machine code containing instructions that a specific hardware processer understands and can execute. Because different types of computer processors execute very different machine language code instructions, the compiled machine language code for one type of computer hardware cannot be executed directly on a different type of hardware computer (for example, machine code for an Intel processor cannot run directly on a PowerPC processor, and vice versa). By interpreting portable Java bytecode instructions, or else converting them into the machine code of the specific processor being used, the JVM enables a single version of bytecode to run on top of different types of hardware and operating systems. In essence, the JVM acts as an abstraction layer for all types of hardware processors.

40. Sun released the first Java Runtime Environment, including the first Java Virtual Machine implementation and the first version of the Java API packages, in 1995. The JRE is designed to execute a Java program's bytecode by providing an abstracted, platform-independent execution environment for the bytecode. To provide an abstracted environment means that the JRE provides

substitute functionality for the functionality that would normally be provided by the physical machine and/or the associated operating system. Platform independence is achieved because the functionality provided by instances of different JREs is consistent across different underlying physical machines and operating systems. In other words, an instance of a JRE is an implementation, for a given hardware and operating system platform, that runs in a process and executes the bytecode compiled from a Java source code program. Once a program's source code has been compiled into bytecode, that program bytecode can be run on any JRE instance compatible with the Java Platform Specification.[14] This portability of programs in Java bytecode form is the hallmark of the platform that underpins the "Write Once, Run Anywhere" philosophy.

3) The Java API Packages

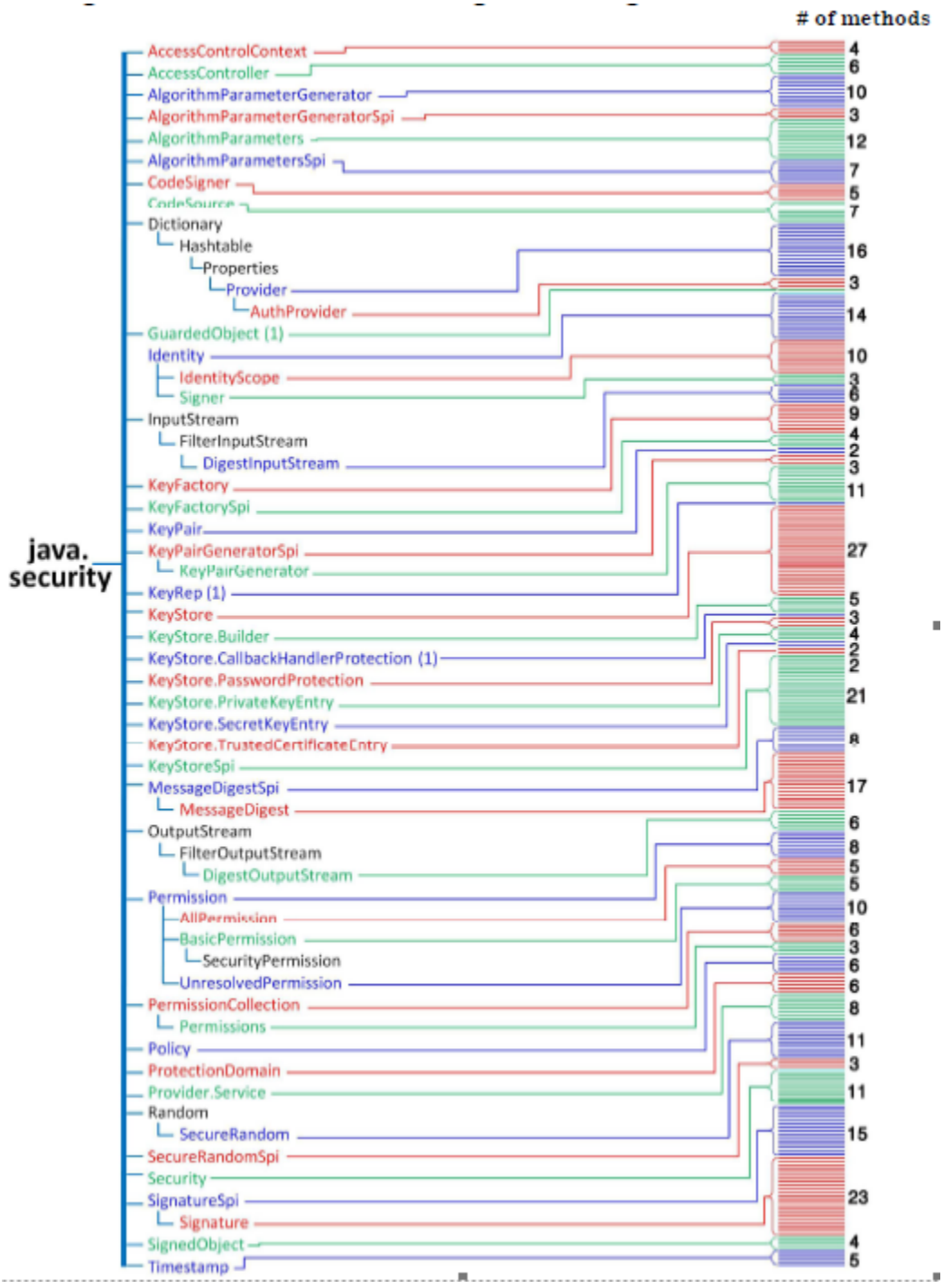
41. Oracle's Java platform contains a collection of resources called the application programming interface packages ("Java API packages"). The relationship of the Java API packages to other parts of the Java platform can be seen in Appendix F – Java SE 5.0 Platform. The Java API packages are prewritten Java programs with classes and methods. Each Java API package has groups of classes and methods that perform discrete operations.[15] These Java API packages provide valuable material that helps developers write applications more quickly. A particularly important benefit of these Java API packages is allowing developers to avoid the tedious and error-prone effort associated with writing their own classes and methods for certain behaviors.

42. A more detailed look at the Java API packages, including examples of API code, the organization of Java API packages, and the way that they are used in the Java platform and by developer or programmers follows.

43. First, the following Figure 2 illustrates, by way of the example of "java.security" API package, how Java API packages include groupings of code called "classes" and, within those classes, pre-written groupings of code called "methods."

[14] The Java Platform Specification includes the Java Language Specification, the Java Virtual Machine Specification, and the Java API Specification.
[15] Java API Specifications, Oracle, http://www.oracle.com/technetwork/java/api-141528.html (last visited January 7, 2016).

Figure 2: Composition of a Java API Package (java.security)

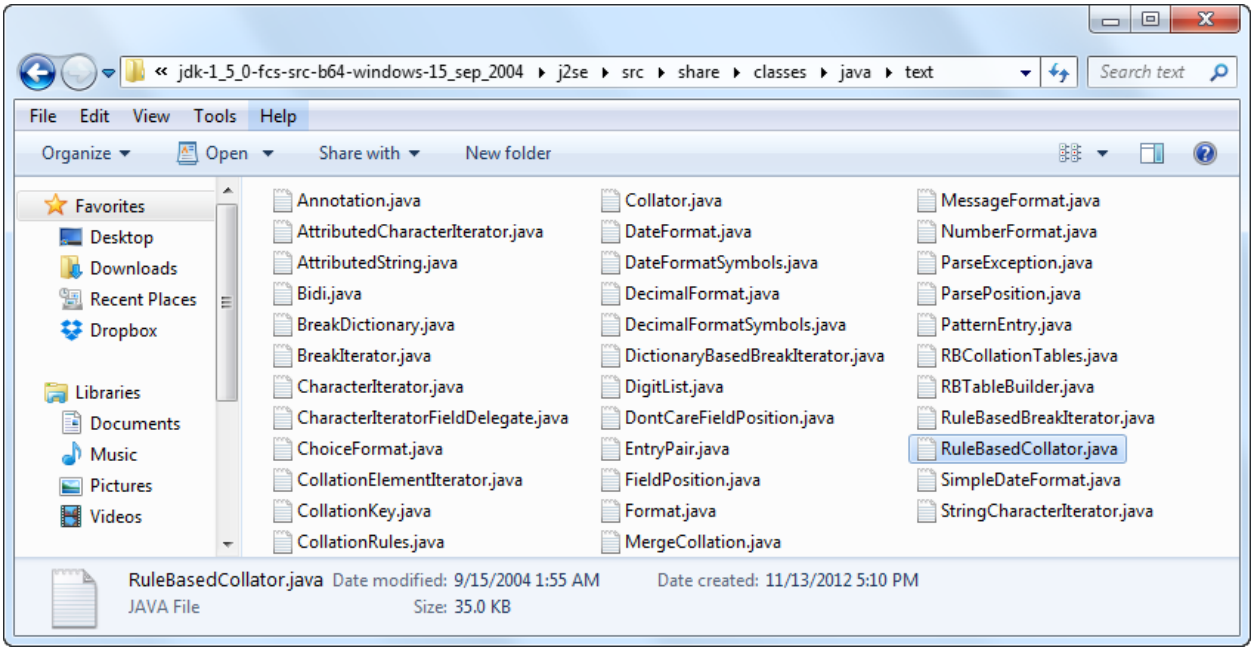


This java.security Java API package, of course, only represents a fraction of the API packages in the Java platform.

44. Once a Java API package has been created, a developer with access to it can use all of the features and capabilities that are provided by its constituent classes and methods. In effect, developers can leverage the pre-written API package without writing it themselves, while only having a minimal understanding of how the API works internally. Developers get the Java API packages within the Java platform via the Java Development Kit ("JDK").

45. The code for the Java API packages exists in different forms. First, the Java API packages exist in the form of source code. In particular, in each Java API package there are collections of classes. The source code that Sun/Oracle created for each class is contained in a source code file with a "*.java*" file extension. For example, Figure 3 below represents the source code files for classes within the Java API package "java.text."

Figure 3: Example of source code files for Java API package "java.text"



The java.text package contains classes and interfaces for handling text, dates, numbers, and messages. Each class is generally stored in a source code file named after the class. For example, the code for the class "RuleBasedCollator" is contained in the file "RuleBasedCollator.java."

46. As demonstrated in the JDK, within a particular .java file, there is typically source code for multiple pre-written methods. For example, an excerpt of the source code for the class "RuleBasedCollator" is reproduced in Figure 4 below.

Figure 4: Excerpt of source code for class "RuleBasedCollator"

```
    public CollationElementIterator getCollationElementIterator(
                                                    CharacterIterator
source) {
```

The source code above is for a method, getCollationElementIterator, which returns an object (instance of a CollationElementIterator) that can be used to walk through a string of text, e.g., in order to compare the characters in two strings or to sort the characters in a string.

47. The code example above has a method named "public CollationElementIterator getCollationElementIterator(CharacterIterator source)." This source code is created by Oracle and it is called declaring code, which is code that identifies, introduces, and specifies a variety of elements, including package statements, import statements, class declarations, interface declarations, field declarations, method declarations, constructor declarations, and annotations[16] (**Error! Reference source not found.** below summarizes a number of the key Java elements).

Table 1: Java key elements

Concept	Definition
Packages	Packages are collections of pre-written code that provide programmers a convenient way to access and share related types (classes, interfaces, enums, and annotations).
Import statements	Import statements specify the packages and classes that are used in code.
Class	A class can be thought of as a grouping of code that includes variables (called "fields") and methods that operate on the variables. Classes can be used for creating programs.

[16] The Court of Appeals stated that: "Declaring code is the expression that identifies the prewritten function and is sometimes referred to as the 'declaration' or 'header.' The expressions used by the programmer from the declaring code command the computer to execute the associated implementing code, which gives the computer the step-by-step instructions for carrying out the declared function." United States Court of Appeals for the Federal Circuit, Oracle America Inc. v. Google Inc. 2013-1021 -1022, p. 9.

Concept	Definition
Interface	An interface is similar to a class. While a class can contain method declarations and the code to implement the methods, an interface can only contain the method declarations but not the code to implement the methods.
Constructor	Java constructors are special methods that are called when an instance of a class is created (i.e., "instantiated"), thereby resulting in a new object of that class. The constructor initializes the newly created object and can set variables to their initial values before any other operations take place.
Method	A method is code that performs a dedicated behavior in an object-oriented programming language like Java.
Field	Variables inside classes are called fields.
Exception	An exception is an event that occurs during the execution of a program that disrupts the normal flow of instructions.
Error	An error is typically a catastrophic event from which a program cannot recover and that usually results in the program terminating. For example, a computer running out of memory is an example of an error.
Enum	An enum indicates a special data type that represents pre-defined values.
Annotation	In Java, annotations are special comments that specify information about classes, methods, variables, parameters, and packages. Whereas comments in source code are ignored, and are removed when the code is compiled, annotations can remain in compiled code to indicate information about that code. Annotations may also cause a development tool to take a different action when compiling or executing the code.

Oracle created this declaring code so that programmers can know in an easy-to-learn way how to use a particular method or other Java element. For example, programmers who use the Java platform come to know that the declaring code for the "RuleBasedCollator" class can be used to invoke the "getCollationElementIterator()" method in the context of the text searching and sorting operations.

48. The modifier "public" determines the level of access to the method by the other classes, in this case, access is available to all who use the java.txt package. The return type of the "geCollationElementIterator()" method is "CollationElementIterator", which declares the type (i.e., class) of object that is output when this method is invoked. "geCollationElementIterator" is

the name of the method, and inside the parentheses is the input parameter for this method, where "CharacterIterator" is the type of input represented by the variable "source".

49. Declaring code is the code that identifies, introduces, and specifies, as well as a variety of other elements, including package statements, import statements, class declarations, interface declarations, field declarations, method declarations, constructor declarations, and annotations.[17]

50. Returning to my discussion of the code in the previous paragraph, there will be additional code after the opening curly brace " { " that carries out the implementation of the "getCollationElementIterator()" method. That additional code is called implementing code.

51. When developers write applications that reuse parts of the prewritten Java API packages, they write source code that refers to the Java API packages' "declaring code." This instructs the Java compiler to refer to bytecode for the Java API packages' prewritten classes and methods. These bytecode of the Java API packages resides in the Java Runtime Environment on the computing device where the developer's program is running.

52. A developer using a software library does not need to have the source code for the Java API packages in the class libraries because the developer has learned to reference Oracle's easy-to-learn declaring code in the Java API packages. Programmers only need to write source code that refers to the declaring code of the Java API packages, which is how developers and end-users use and benefit from the Java API packages' declaring code and the organization it defines for the underlying classes and methods.

53. The structure, sequence, and organization (SSO) includes the taxonomy of the packages, classes, methods, constructors, fields, interfaces, errors, exceptions, enums, and annotation types defined in the Java API packages. The SSO also includes the hierarchy, grouping, intricate and

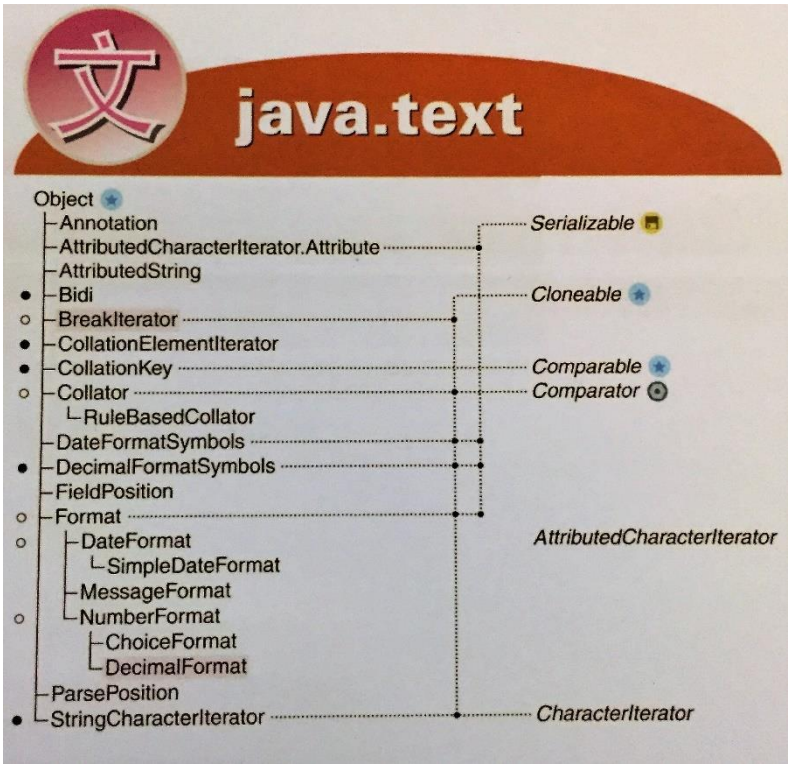
[17] The Court of Appeals stated that: "Declaring code is the expression that identifies the prewritten function and is sometimes referred to as the 'declaration' or 'header.' The expressions used by the programmer from the declaring code command the computer to execute the associated implementing code, which gives the computer the step-by-step instructions for carrying out the declared function." United States Court of Appeals for the Federal Circuit, Oracle America Inc. v. Google Inc. 2013-1021 -1022, p. 9.

complex relationships, inheritance, and implementation of interfaces between and among the foregoing elements and portions of those elements.[18]

54. Oracle invested in developing many thousands of these code declarations, and organizing the relationships between and among them in a creative and easy-to-learn way within the classes and packages. Oracle engineers expressed a very particular structure, sequence and organization of these methods, classes, interfaces, etc. For example, Oracle chose to bundle the following classes (and interfaces) (each containing a number of specifically selected methods) within the package "java.text": AttributedCharacterIterator, CharacterIterator, Annotation, AttributedCharacterIterator.Attribute, AttributedString, Bidi, BreakIterator, ChoiceFormat, CollationElementIterator, CollationKey, Collator, DateFormat, DateFormat.Field, DateFormatSymbols, DecimalFormat, DecimalFormatSymbols, FieldPosition, Format, Format.Field, MessageFormat, MessageFormat.Field, NumberFormat, NumberFormat.Field, ParsePosition, RuleBasedCollator, SimpleDateFormat, and StringCharacterIterator. I depict this structure, sequence, and organization in Figure 5 below.

Figure 5: java.text

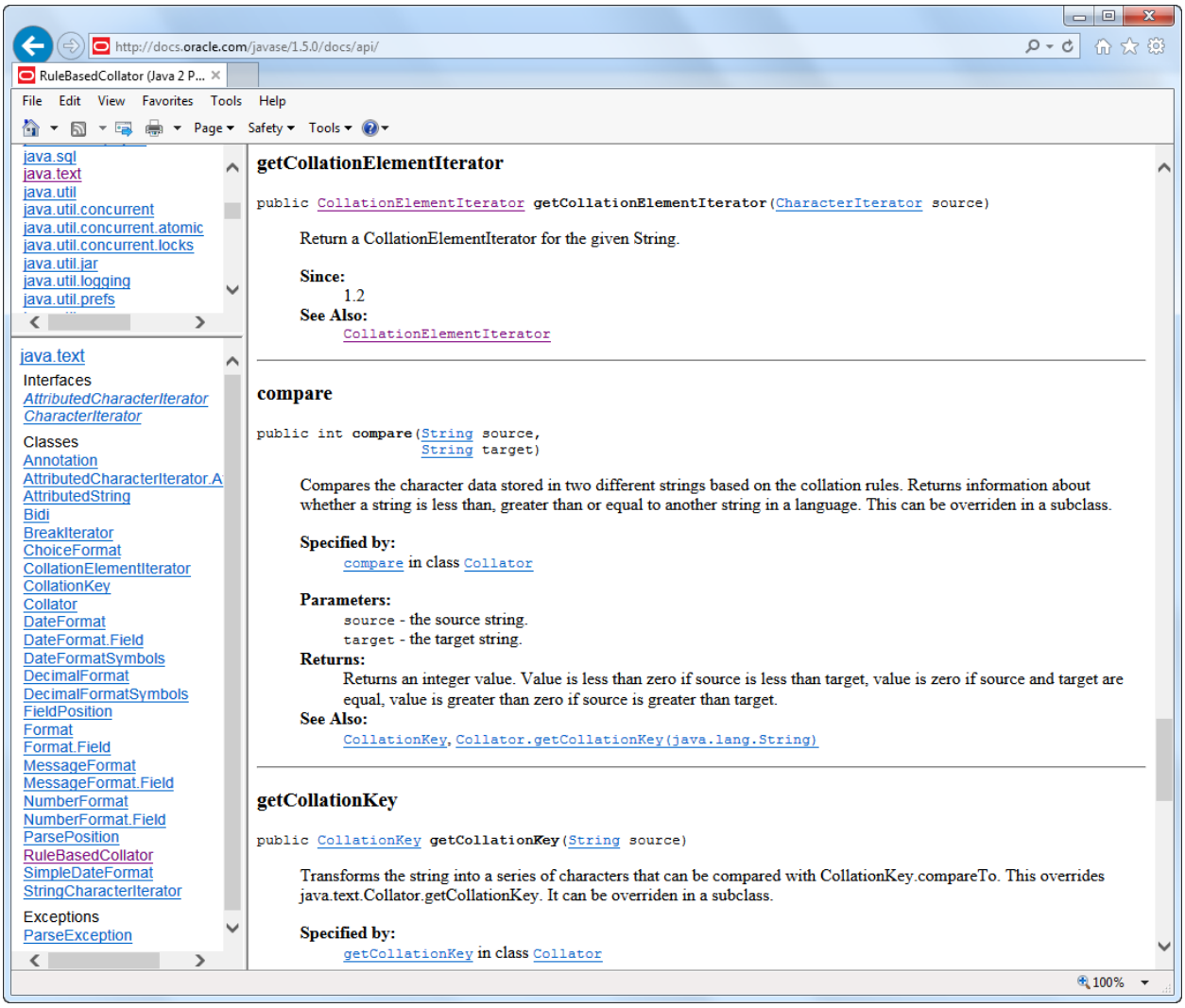
[18] The Court of Appeals stated that: "Google copied the elaborately organized taxonomy of all the names of methods, classes, interfaces, and packages—the 'overall system of organized names—covering 37 packages, with over six hundred classes, with over six thousand methods.' Copyrightability Decision, 872 F. Supp. 2d at 999. The parties and district court referred to this taxonomy of expressions as the 'structure, sequence, and organization' or 'SSO' of the 37 packages." United States Court of Appeals for the Federal Circuit, Oracle America Inc. v. Google Inc. 2013-1021, 1022, p. 11.



55. The Java API Specification describes the Java API packages, including its classes and methods. [19] For example, the Java SE 5 API Specification is located at the url: https://docs.oracle.com/javase/1.5.0/docs/api/ (for a list of all packages as well as copied packages in Java SE 5, see Appendix G – Packages in Java SE 5.0 and Copied API Packages). The Java API specification documentation is created using a tool called Javadoc that is run on Java source code files (ending in .java) and: (1) extracts from the file the declaring code and also comments in the code, and (2) organizes that code and associated comments in a way that shows the declaring code itself, as well as reflects the hierarchical structure, sequence, and organization of the methods within classes, and the classes within the Java API packages. Developers access the Java API Specification and become familiar with the Java API package's expression from that documentation and from use of the source code. For example, Figure 6 below shows a portion of the Java API Specification document reproducing the declaring code defined by the Java API package for the "getCollationElementIterator()" method:

[19] Java API Specifications, Oracle, http://www.oracle.com/technetwork/java/api-141528.html (last visited January 7, 2016).

Figure 6: Example of the declaring code for the Java API package for the "getCollationElementIterator()" method



The declaration in this documentation shows how the getCollationElementIterator() method fits within the larger hierarchical design of the structure, sequence, and organization of methods, method declarations, classes and packages (and has text descriptions describing the code).

B. Background on the Android Platform

56. Google generally announced its Android programming platform in 2007. In September 2008, Android was released commercially.[20] Android now runs on many devices, from mobile phones to wearables. The Android Platform includes the Android Runtime as shown in Appendix H – Android Platform. The Android Runtime includes "Core Libraries" as shown (which contain the 37 APIs), as well as the Dalvik/ART virtual machines (for a list of copied Java packages, see Appendix I - List of Copied Oracle APIs).

1) The Android API (including the 37 Java APIs)

57. Google's Android programming platform provides development tools, including API packages, for writing Android apps. These APIs have gone through multiple version releases that have numbers and names as shown in Appendix J - Android API Levels and Versions. Those packages include the 37 Java APIs. Android offers additional API packages that Google developed. Android has a similar structure to the Java platform's developer tools and their provision. For example, (1) Android API packages are also programs with prewritten classes and methods, and are used by developers to build apps, (2) developers access these Android API packages through an Android Software Development Kit (SDK), and (3) developers on the Android platform can use the Android API packages to build similar apps. In sum, Google's use of Android API packages including the 37 Java APIs is similar to how these packages are used on the Java platform.

58. As with the Java platform, the Android API packages exist in source code format (.java files), and are written in the Java programming language. The Android API packages containing the pre-written classes and methods are distributed by Google in compiled form within the Android Runtime (the Android API packages are known as the core libraries). When programs for Android have been written with code referencing the Android API packages in the Android Runtime, they make calls upon the Android API packages (class libraries) to invoke the pre-written programs, classes, and methods within the Android API packages.

[20]Dan Morrill, *Announcing the Android 1.0 SDK, release 1*, Android Developers Blog (Sep. 23, 2008), http://android-developers.blogspot.com/2008/09/announcing-android-10-sdk-release-1.html.

59. As with the Java API packages, the Android API packages are organized with collections of methods, organized within classes, and the classes are organized within packages. Programmers developing apps in the Android platform use the Android API packages in the same way that they use API packages in the Java platform. Moreover, the Android API packages, classes and methods perform the same functions in Android applications as the Java API packages perform in Java applications. As discussed in more detail below, for the 37 Java APIs that are also contained in the Android API packages, the structure, sequence, and organization are the same.

60. So, for example, as discussed above, in Android there is an API package called "java.text," that contains a class called "RuleBasedCollator" (within the file "RuleBasedCollator.java"), that in turn contains a method "getCollationElementIterator()" that is declared in precisely the same way as in the Java package:

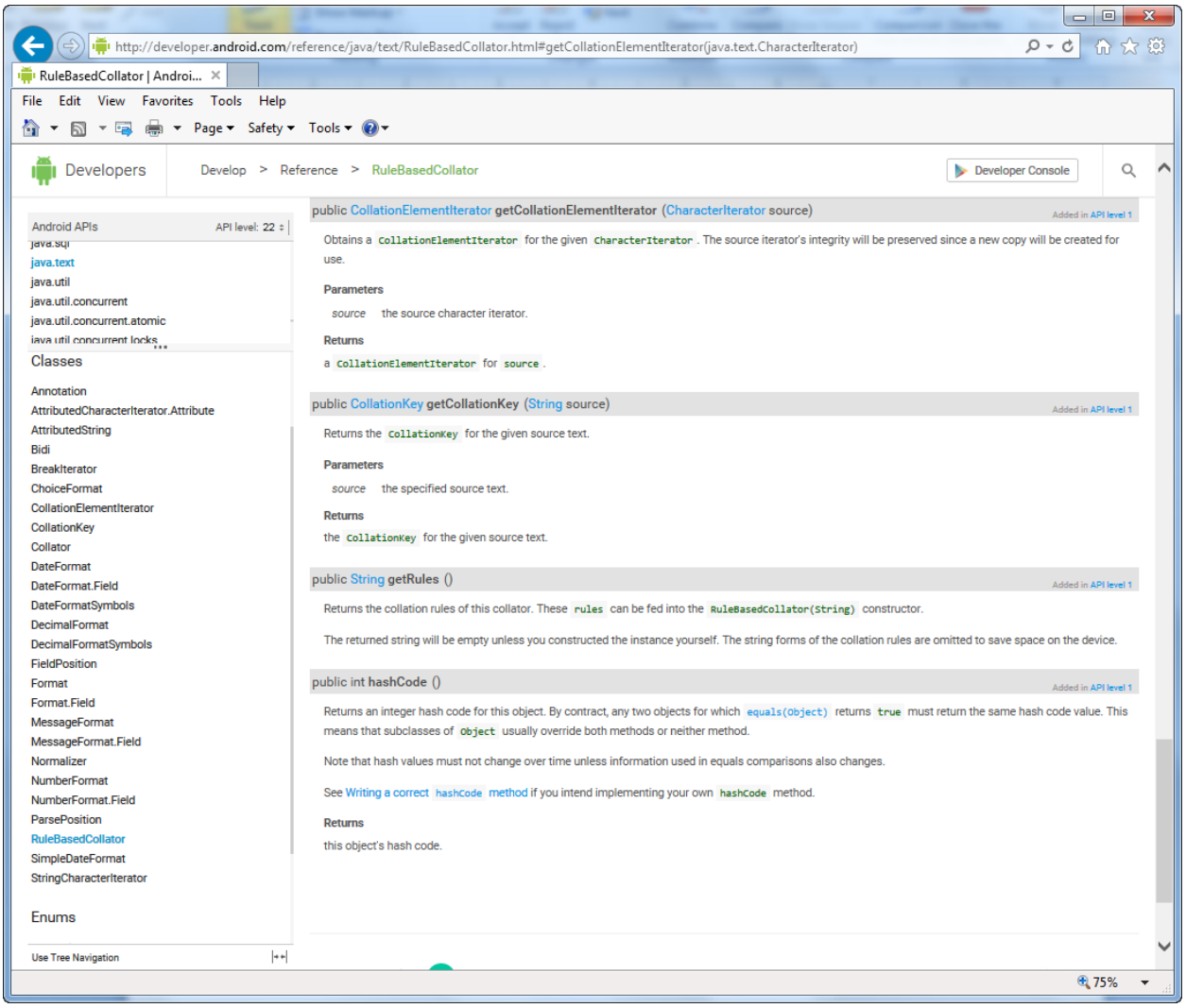
```
    public CollationElementIterator getCollationElementIterator
(CharacterIterator source) {
```

This is just one of many examples showing how the declaring code from the 37 Java APIs is used in the same way, in the same organization of methods and classes, to invoke the same method within the Android platform.

61. In Android, the API packages and their constituent classes and methods are set forth in a particular, discrete Android API specification document labeled Android Packages. For example, the Android API Specification is located here: http://developer.android.com/reference/packages.html (and different API *levels* corresponding to different versions of Android can also be seen there). The Android API Specification Documentation is created in a similar way to the Java Specification documentation. In particular, a tool is used to analyze the .java source code files and extract from the code the declaring code and also comments in the code. This tool organizes the extracted code and associated comments in a way that shows the declaring code itself, as well as reflects the hierarchical structure, sequence, and organization of the methods within classes, and classes within the API packages. The API Specification for Android is often how developers come to learn the API declaring code that is contained in the API package source code files.

62. Figure 7 below show the portion of the Android API Specification document that reproduces the declaring code defined in the Android API package for the "getCollationElementIterator()" method.

Figure 7: Portion of API Specification Document in Android reproducing declaring code for the Java API Package for the "getCollationElementIterator()" method



This documentation shows how the getCollationElementIterator() method fits within the larger hierarchical design of the structure, sequence, and organization of methods, method declarations, classes and packages (and has text descriptions describing the code).

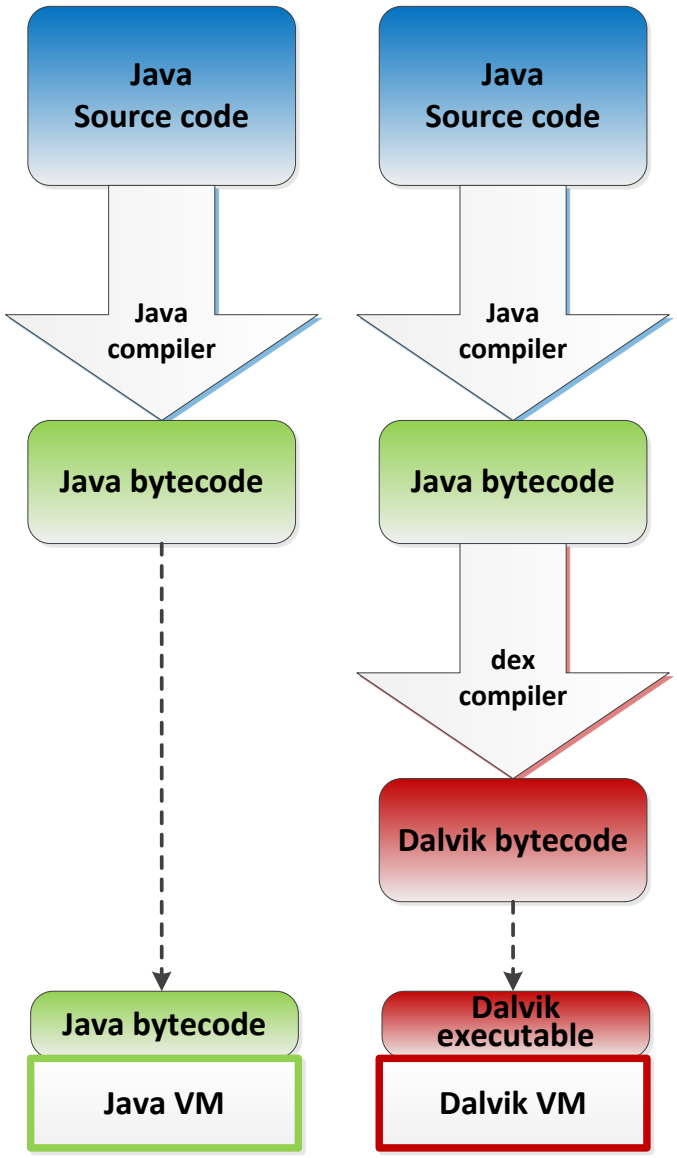
2) The Android Dalvik/ART Virtual Machines

63. Android has a virtual machine that is different than the virtual machine in Java. The original Android virtual machine is called the Dalvik virtual machine. In Android "Lollipop" (Version 5.), released in November 2014, Android contained a default virtual machine called the Android Runtime (or "ART") virtual machine[21] that is also different from the Java virtual machine. As with the Java platform, the Android virtual machines are part of the Android platform and are distributed on devices along with compiled versions of the API packages, including the Java APIs; collectively these components are called the Android Runtime. When the Android Runtime is present on a computer or device, then an app written for the Android platform can be executed and run.

64. The execution of an Android app in the Android virtual machine operates as follows. First, because apps written in Android are lines of code written in the Java language, they are compiled into an intermediate representation called "Java bytecode," i.e., .class files of the general type that might be found in Java. However, Android then converts and further compiles the .class files into an incompatible form of bytecode called "Dalvik" bytecode, which are files ending with the extension .dex. This bytecode is specific to Android and is designed to run only on Android's Dalvik and ART virtual machines. The diagram in Figure 8 compares how source code for Android is ultimately converted into Dalvik bytecode, as opposed to Java bytecode.

[21] Ron Amadeo, *Android 5.0 Lollipop, thoroughly reviewed,* ars technica (Nov. 12, 1214), http://arstechnica.com/gadgets/2014/11/android-5-0-lollipop-thoroughly-reviewed.

Figure 8: Representation of source code for Android converted into Dalvik bytecode



As discussed in more detail in Section IX.B below, because the Java bytecode created from the source code in Android is then compiled into Dalvik bytecode and .dex files, the ultimate resulting Android programs cannot run on the Java platform because the Dalvik bytecode is not compatible with the Java Virtual Machine.

VI. THE JAVA PLATFORM HAS FOSTERED A ROBUST, DYNAMIC COMMUNITY OF DEVELOPERS

A. Sun Microsystems developed the Java platform to support an object-oriented approach to programming

65. Beginning in the early 1990s, Sun Microsystems developed an approach to programming that would be appropriate for the simultaneous expansion of computing devices, networks, and application development.[22] The result was the family of innovations encompassed in the Java platform that today includes the Java language, virtual machine, and libraries of application programming interface (API) packages.

66. From the perspective of software developers, the Java API packages represent expression of the following benefits: they are object-oriented; they can be implemented across distributed networks; they are resistant to numerous glitches and inefficiencies; they are portable; and, they are architecture-neutral.[23]

67. From a practical programming perspective, Java API packages offer easy-to-learn expressions to invoke pre-written programs that developers can reuse instead of having to engage in otherwise tedious code-writing. Rather than requiring developers to write entire code sequences to perform specific tasks from scratch, the Java API packages offer a compatible means to deploy packaged bundles of object-oriented code classes and methods to achieve complex behaviors.

68. Java's API structure, sequence, and organization of classes, methods, interfaces, and other component interactivity offers developers approachable options for writing applications.

[22] The History of Java Technology, Oracle, http://www.oracle.com/technetwork/java/javase/overview/javahistory-index-198355.html (last visited January 7, 2016).

[23] James Gosling, *Java: An Overview,* Dartmouth (Feb. 1995), pp. 2-5 *available at:* http://www.cs.dartmouth.edu/~mckeeman/cs118/references/OriginalJavaWhitepaper.pdf.

The Java API packages are well designed, coherent, and easy to learn. The Java API packages' stylizations of class and method arrangements, and elegant approach to application building makes them valuable to developers and the applications they create; and, by extension, to end-users of many computing devices running those applications.

69. For these and other reasons, the Java platform has been widely adopted.

B. Compatibility of the Java API Packages is critical to sustaining Java's value proposition to developers

70. Sun and Oracle have consistently emphasized the importance of compatibility, as reflected in their "Write Once, Run Anywhere" philosophy, which they have realized via published specifications of the Java platform (including the Java language and virtual machine,[24] as well as the Java API packages)[25] and Technology Compatibility Kit (TCK) test suites that check particular implementations of a Java Specification for compliance. Oracle has used its leadership and stewardship of the Java platform to ensure that the developer community can continue to benefit from the Java API packages.

71. Compatibility of the Java API packages is beneficial to developers since it allows them to develop applications that run without change in a wide range of operating environments (consisting of different operating systems and hardware versions), thereby simplifying the effort required to evolve their applications to meet new customer requirements and business opportunities.[26]

---

[24] Java Language and Virtual Machine Specifications, Oracle, https://docs.oracle.com/javase/specs/ (last visited January 7, 2016).

[25] Java API Specifications, Oracle, http://www.oracle.com/technetwork/java/api-141528.html (last visited January 7, 2016).

[26] "Java is the programming language of choice for 9 million developers and today powers 7 billion devices. Improving road and air safety, collecting information from the world's oceans for science applications, increasing grain crop quality and quantifying to help feed the hungry, simulating the human brain and musculoskeletal system, and gaming are some of the intriguing projects worldwide that use the Java technology. Enterprise developers can choose from an ecosystem of 30 Java EE 6 and Java EE 7 compatible implementations from 12 vendors. Additionally, more than 125 million Java-based media devices have been deployed and over 10 billion Java Cards have been shipped since Java's introduction." (http://www.marketwired.com/press-release/oracle-and-the-community-celebrate-20-years-of-java-nyse-orcl-2021578.htm)

72. Compatibility of the Java API packages also helps end-users. It gives them more choice and ability to match apps to their devices according to their business requirements and personal preferences, and they can also enjoy the productivity and creativity of a vibrant development community that can work together.[27] Additionally, end-users benefit from enhanced operational robustness because compatibility means critical applications do not break when ported to new operating environments.[28]

73. Compatibility of the Java API packages also helps educators. It allows them to teach key concepts, principles, and features of the Java platform so their students' knowledge will be consistent throughout their schooling and transferrable to their careers. Conversely, the lack of compatibility between the Java platform and Android platform is confusing to both students and educators. For example, as a Professor of Computer Science who regularly teaches courses on both the Java platform and the Android platform, it is pedagogically frustrating because when we are covering the Java portion of my classes I introduce the powerful features of Java 8 (such as Lambda expressions and the Streams API), but then when I switch to Android the students have to forget what they learned about Java 8 to stay within the constraints of the Android platform.

74. Compatibility is ultimately necessary for the constructive evolution and growth of the Java platform. When programmers support an ecosystem by providing compatible devices and code, they grow the available pool of applications, additional API packages and devices, thereby further incentivizing other developers to contribute to the platform.[29]

---

[27] "'Java directly or indirectly touches just about every human on this planet. It is the glue that allows mobile health and banking in remote areas of the world, entertains millions with games such as Minecraft, and drives the economic engine of our global markets,' said Martijn Verburg, co-leader of Java User Group, London. 'It is extremely important that this platform be managed by a leader who understands Java's importance to the world, and Oracle does an excellent job in collaborating with a hugely diverse community of users, developers, open source projects and larger enterprises.'" (http://www.marketwired.com/press-release/oracle-and-the-community-celebrate-20-years-of-java-nyse-orcl-2021578.htm)

[28] "'Java has grown and evolved to become one of the most important and dependable technologies in our industry today. Those who have chosen Java have been rewarded many times over with increases in performance, scalability, reliability, compatibility, and functionality,' said Georges Saab, vice president of development, Java Platform Group at Oracle. 'The Java ecosystem offers outstanding libraries, frameworks, and resources to help programmers from novice to expert alike.'" (http://www.marketwired.com/press-release/oracle-and-the-community-celebrate-20-years-of-java-nyse-orcl-2021578.htm)

[29] "[D]evelopers each contribute to making the platform better—because when developers support a platform with their applications, the platform becomes better and more attractive to consumers," https://source.android.com/compatibility/

VII. GOOGLE CHOSE TO BASE THE ANDROID PLATFORM ON THE 37 JAVA APIS BECAUSE OF JAVA'S KEY ATTRIBUTES

75. Java has been one of the most popular and widely used development and runtime platforms for the past 15 years, due in large part to Sun's and Oracle's focus on "Write Once, Run Anywhere" compatibility covered in Section IX.B. As a result, many application developers know Java and its API packages. It is also one of the most commonly taught platforms in schools and universities around the world. Moreover, many tools are available for writing, refactoring, debugging, optimizing, and deploying Java applications. This familiarity with an ecosystem enhances the productivity of developers on the Android platform.

76. The declaring code and organization of the Java API packages developed by Sun—as well as the complex interactive relationships between these packages—were created over time in a deliberate and creative manner designed to maximize this appeal, usability and efficiency that I describe above. The challenge and complexity of developing creative API packages can be seen in a number of projects in the Java platform ecosystem dating back to the late 1990s.

77. For example, Java ACE is an object-oriented concurrent network programming middleware project that I led starting in 1996 together with some of my graduate and undergraduate students at Washington University, St. Louis.[30] Java ACE was an academic, non-commercial research project focused on understanding the strategies, tactics, and patterns needed to develop middleware that effectively supported concurrent and networked applications within the context of the Java 1.0, 1.1., and 1.2 platforms.[31] From June 1996 to August 1999, myself and my team of students devoted significant effort and creativity developing the Java ACE toolkit, which contained roughly 20,000 lines of Java code and about a dozen packages. It took us this amount of time to express the functionality in Java ACE.[32]

---

[30] *See* Java ACE, Vanderbilt, http://www.dre.vanderbilt.edu/JACE/ (for access to Java ACE downloads).

[31] A paper describing our experience creating Java ACE appears at http://www.dre.vanderbilt.edu/~schmidt/C++2java.html.

[32] ObjectSpace is another example of the effort and complexity involved in creating the types expressions in the Java API packages. In 1997, ObjectSpace, Inc. released its Java Generic Library (JGL). Whereas Sun's Java Collections Framework released with JDK 1.2 in 1998 were arranged into packages of classes and methods, the JGL is organized based on "containers" of data structures, using design patterns guided by the C++ Standard Template Library (STL). JGL is more than simply one different arrangement of Java code—it is a gateway to vast diversity of Java utilization. An article from the time period explains, "The combinations are almost endless. You may think they are far from infinite because the roughly 20 types of Containers, multiplied by roughly 40 algorithms, produces some 800 different

VIII. WITHOUT THE COPIED ORACLE DECLARING CODE AND API PACKAGES, ANDROID WILL NOT COMPILE AND IS RENDERED INOPERABLE

78. Google copied the 37 Java APIs in a manner that renders the operation of the Android platform critically dependent on each API package at issue. I find that Android is not usable on a computing device, such as a phone or tablet, without each of the Java APIs packages at issue or the copied declaring code in them. Regardless of the size of the Java API code lines in the operating system profile, their removal is fatal to Android's operability. Just as the brain stem, heart or pancreas represent fractional shares of a human being's overall body mass, their surgical removal would clearly be of far greater consequence than the amputation of an entire limb.

A. Background on the Android build process

79. The Android build process broadly refers to the compiling and assembling of the source code files, such that they are ready to run on an actual Android device. The build is the creation of the actual Android platform from the source code.

80. After a build is completed, system image files are produced. As I explained earlier, machine code is code that can be executed. When I talk about the system image file being produced, I am talking about the source code for the Android platform being compiled into machine code. System image files are then 'flashed' onto a device. Flashing refers to the process by which the code of the Android platform is stored on the device. If the device successfully boots a working version of Android from these images, it means the build was successful. An image that fails to boot on the device, or a build that does not produce these images, is treated as a failed build.[33]

---

combinations. In fact, this number ends up looking very small when you factor in the extra dimension of function objects that can customize most of the algorithms."

See: Laurence Vanhelsuwé, *Need a good set of abstract data structures? ObjectSpace's JGL packs a punch!,* JavaWorld (Jan. 1, 1997), http://www.javaworld.com/article/2076958/java-app-dev/need-a-good-set-of-abstract-data-structures--objectspace-s-jgl-packs-a-punch-.html.

[33] The system.img, userdata.img, and boot.img files are sufficient to launch Android when flashed onto a device, and the system.img image file is necessary to start Android on a device. As a result, these are the three image files that will be searched for once a build test has completed.

81. Performing a build involves a widely known, standardized protocol that consists of three major steps. First, an appropriate build environment must be set up, which includes downloading several utility packages, defining environment variables, and setting the specific configuration of the build. Second, the source code for the appropriate version of Android must be downloaded (e.g., from the online Android Open Source Project (AOSP) code repository).[34] Third, the makefile must be started, which automatically performs the build steps (a makefile describes the dependencies necessary to map source code to machine code).

82. With this understanding of the build process, I have conducted several tests to understand the extent to which the Android build process depends on the Java APIs. I describe these tests in the next sub-section.

B. Methodology of the Build Tests

83. I assess the dependence of Android mobile devices on the 37 Java APIs by recreating the operation of an Android device through a systematic procedure called a "build test." Put simply, a build test attempts to create the Android platform from source code and evaluates whether Android performs as it would in a production environment. By removing certain elements from the Android build, I am able to test the dependency of the Android stack on individual components, from groupings of API packages, to individual APIs packages, to individual segments of code.

84. The tests described here use Android 5.1.1, release 30 (Lollipop, API Level 22). One important consideration in ensuring consistency across all tests involves verifying that the build environment in every case is exactly the same.

85. To this end, the tests were conducted on a remote server (Amazon AWS EC2 servers running the Ubuntu 14.04 operating system as a test environment). All server instances were the same to ensure the build environment used in every test was identical. In each different test, the build was attempted to see if it was able to complete, subject to the modifications that were

[34] *See* Android Open Source Project, https://source.android.com/, (last visited January 7, 2016).

made in that particular experiment.[35] No material from the source code other than the copied 37 API packages was modified in any way during any of these tests.

86. As a control case for the tests, a single successful build of the Android platform was performed using a server instance generated from this build environment. This control case verified that the environment used for the tests produces a successful build when the source code is left unaltered.

C. Results

87. I find that the 37 Java API packages at issues and their copied declaring code are required for the build process to complete successfully. Accordingly, they are required for the functioning of the Android platform.

88. I performed three different types of tests to understand and corroborate the robustness and breadth of my findings. I describe the methodology and the results of each test in detail in this sub-section.

1) Android will not function when all copied 37 Java API packages at issue are removed

89. The first build test involved attempting the build process after removing all source files belonging to the copied 37 Java API packages at issue from the Android source code. The actual directories in which the source files are located were not deleted - only the source files inside them were removed. Once this material was removed, the build was attempted to see if it was able to complete successfully without the material from the 37 Java API packages at issue. During this attempt, the build failed in the following way – the build returned an error and aborted. Appendix K – Error Log from Android Build Test No. 1: Removal of the Entire Set of 37 Copied APIs shows the full error log for this build attempt.

90. Verifying the results through the record of the build contained in the 'out' folder revealed that building without the copied 37 Java API packages at issue failed to produce the image files that are required to run Android on a device. This result indicates that Android fails to build

[35] The source files from the 37 Java API packages at issue that are modified in this experiment are found in the libcore/luni/src/main/java folder of the source code root directory.

successfully when the 37 Java API packages at issue are removed. This failure to build illustrates that Android is unusable without the presence of the 37 Java API packages at issue.

2) Android will not function if any one of the copied 37 Java API packages at issue is removed

91. The second set of build tests attempted the Android build when each of the copied 37 Java API packages at issue was individually removed from the source code. This test was repeated 37 different times, each time removing a different copied API package. For every test attempted, the build aborted after returning an error. Appendix L - Error Log from Android Build Test No. 2: Removal of Individual API Source Files, shows the error logs for the 37 repetitions.

92. Searching through the build output once again showed that the system images that are needed to boot Android on a device were not created; this same result was observed for all 37 iterations. Together, these results show that Android fails to build successfully when any single one of the 37 copied API packages at issue is removed. These 37 failures to build illustrate that Android is unusable in the absence of any one the 37 Java API packages at issue.

3) Android will not function if the copied lines of copied declaring code are deleted from the source files of the copied 37 Java API packages at issue

93. The third and final test measured the dependence of the Android build process on the individual lines of copied declaring code inside each of the 37 Java API packages at issue. To this end, the build was attempted after identifying and deleting these declaring lines from the source files of the 37 Java API packages at issue from the source code.

94. It is important to note that this process did not involve deleting any actual source files; only the specific lines of code from inside the source files that were classified as copied declaration lines were removed. This removal was accomplished using the results of a prior analysis by Mr. Zeidman that identifies which particular lines in which source files constitute copied declarations. In particular, I have reviewed the expert report of Mr. Zeidman including Exhibit S of that report in which he lists the declaring code copied by Google. It is this copied declaring code that I removed from the source code files of the 37 packages in order to conduct my tests. Android will not function if the copied lines of declaring code are deleted from the source file of any copied class.

a) Android will not function if the copied lines of declaring code are deleted from the source files of a single class of the copied 37 Java API packages at issue

95. As a first test, I took all of the copied declaring code out of the class java.text.annotation while leaving all other code intact. I then attempted to build the Android platform and it failed. When the build was attempted without each of the copied declarations, the build process returned an error and aborted prematurely every time. The complete error log summarizing the output of this build attempt can be found in Appendix M - Error Log from Android Build Test No. 3A: Removal of Source File Declaring Codes from java.text.annotation.

b) Android will not function if the copied lines of declaring code are deleted from the source files of the copied 37 Java API packages at issue

96. Next, I performed a final test by removing the copied declaring codes from all of the copied 37 Java API packages at issue. Once again, the output of the build for this test did not contain the necessary system image files. The findings from this test demonstrates that Android fails to build successfully when the copied declaring code lines are specifically removed, while leaving all other non-copied lines of code intact. These failures to build illustrate that Android is inoperable in the absence of any one of the copied declaring codes from inside the copied Java APIs. The complete error log summarizing the output of this build attempt can be found in Appendix N - Error Log from Android Build Test No. 3B: Removal of all copied declaring codes from all 37 APIs.

IX. THE ANDROID PLATFORM VIOLATES THE FUNDAMENTAL "WRITE ONCE, RUN ANYWHERE" PARADIGM OF THE JAVA PLATFORM

A. The Android Platform is incompatible based on the Java Compatibility Kit

97. I understand from the declaration of Mark Reinhold, Chief Architect of the Java Platform Group at Oracle, that Google's implementation of the 37 Java APIs in Android has failed to pass the official compatibility tests which are based on the Java Specifications.

98. The Java Compatibility Kit (JCK) is Java's official test suite for determining compatibility. It is used by Oracle to confirm compatibility of alleged implementations of Java with the adopted

specifications of the JCP.[36] I understand that the Java SE 6 platform passes all of the tests of the Java SE 6 Specification JCK, and that Java SE 7 platform passes all of the tests of the Java SE 7 Specification JCK. I also understand that failure of any one JCK test means that the alleged implementation is not compatible with the corresponding Java SE Specification.

99. A key test within the JCK is the Signature Test, which uses static analysis to determine whether or not an alleged implementation of 37 Java API contains the precisely correct number of classes, methods, interfaces, and fields. The test outputs a listing of missing and/or added class, constructor, field, method, nested class, super class, interface, and annotation elements, each of which constitutes an error. I understand from Dr. Reinhold's Declaration that any errors qualify an implementation as incompatible, i.e., any missing or added element renders the implementation incompatible. In particular, any missing element indicates the alleged implementation of Java represents a "subset" of the Java platform. Likewise, any added element indicates the alleged implementation of Java represents a "superset" of the Java platform.

100. It is my understanding that Dr. Reinhold used the JCK Signature Test to evaluate the compatibility of Android API Level 9 (Gingerbread), released in 2011, with Java SE 6, which was released near the end of 2006. Specifically, he ran the Java SE 6 JCK Signature Test on a version of the Java SE 6 runtime using Android Gingerbread's implementation of the 37 Java APIs.

101. It is my understanding that Dr. Reinhold also used the JCK Signature Test to evaluate the compatibility of Android API Level 21 (Lollipop), released in 2014, with Java SE 7, which was released near the end of 2011. Specifically, he ran the Java SE 7 JCK Signature Test on a version of the Java SE 6 runtime using Android Gingerbread's implementation of the 37 Java APIs.

102. Dr. Reinhold found that the Android Gingerbread's implementation of the 37 Java APIs in the Java SE 6 runtime failed to pass the Java SE 6 TJCK, generating a total of 409 errors for missing or added signatures. The results of this analysis are presented in Table 2 below, organized by type of error and package name.

[36] *See* https://jcp.org/en/resources/tdk, (last visited January 7, 2016), for a page on the Java Community Process (JCP) website describing how Specification Leads for JSRs obtain access to JCK testing tools; *See* http://openjdk.java.net/groups/conformance/docs/JCK6bUsersGuide/JCK6b_Users_Guide.pdf, (last visited January 7, 2016), for a copy of the user's guide for version 6b of the Java Compatibility Kit.

Table 2: Summary of error types for the 37 copied API packages for "addition" and "removal" errors from the Java SE 6 TCK against Android Gingerbread

	Error Type						
	Class		Method		Other		
Package Name	**Added**	**Missing**	**Added**	**Missing**	**Added**	**Missing**	**Total**
java.awt.font	0	1	0	5	0	0	**6**
java.beans	1	1	2	2	0	0	**6**
java.io	0	0	6	1	2	0	**9**
java.lang	2	2	37	12	7	4	**64**
java.lang.ref	1	0	1	0	2	0	**4**
java.lang.reflect	5	2	8	0	10	0	**25**
java.net	5	2	32	5	18	0	**62**
java.nio	2	0	10	0	2	0	**14**
java.nio.channels	0	5	0	28	0	1	**34**
java.nio.channels.spi	0	0	0	1	0	0	**1**
java.nio.charset	2	0	1	0	0	0	**3**
java.security	0	0	2	2	0	0	**4**
java.security.cert	0	2	3	1	0	1	**7**
java.security.spec	0	0	2	0	0	0	**2**
java.sql	0	0	0	14	44	0	**58**
java.text	0	0	5	2	1	0	**8**
java.util	5	3	21	22	3	3	**57**
java.util.jar	1	0	0	0	1	0	**2**
java.util.logging	0	0	1	1	0	0	**2**
java.util.prefs	1	0	1	0	0	0	**2**
java.util.regex	1	0	0	1	0	1	**3**
java.util.zip	4	0	2	0	0	5	**11**
javax.net.ssl	2	0	7	10	2	0	**21**
javax.sql	0	0	0	4	0	0	**4**
Total	**32**	**18**	**141**	**111**	**92**	**15**	**409**

103. Likewise, Dr. Reinhold found that the Android Lollipop's implementation of the 37 Java APIs in the Java SE 7 runtime failed to pass the Java SE 7 TCK, generating a total of 303 errors for missing or added signatures. The results of this analysis are presented in Table 3 below, organized by type of error and package name.

Table 3: Summary of error types for the 37 copied API packages for "addition" and "removal" errors from the Java SE 7 TCK against Android Lollipop

	Error Type						
	Class		Method		Other		
Package Name	**Added**	**Missing**	**Added**	**Missing**	**Added**	**Missing**	**Total**
java.awt.font	0	19	0	5	0	0	**24**
java.beans	1	35	1	1	0	0	**38**
java.io	0	0	0	1	0	0	**1**
java.lang	0	7	0	12	0	4	**23**
java.net	0	6	0	5	0	0	**11**
java.nio.channels	0	23	0	28	0	1	**52**
java.nio.channels.spi	0	1	0	1	0	0	**2**
java.security	0	3	2	2	0	0	**7**
java.security.cert	0	8	0	2	0	1	**11**
java.sql	0	1	0	14	20	0	**35**
java.text	0	0	2	2	0	0	**4**
java.util	5	3	9	22	1	3	**43**
java.util.logging	0	0	1	1	0	0	**2**
java.util.regex	0	0	0	1	0	1	**2**
java.util.zip	0	0	0	0	0	5	**5**
javax.crypto.spec	0	0	0	0	1	0	**1**
javax.net.ssl	0	2	0	10	0	0	**12**
javax.security.auth	0	3	0	0	0	0	**3**
javax.security.auth.callback	0	6	0	0	0	0	**6**
javax.security.auth.login	0	14	0	0	0	0	**14**
javax.security.auth.x500	0	1	0	0	0	0	**1**
javax.sql	0	2	0	4	0	0	**6**
Total	**6**	**134**	**15**	**111**	**22**	**15**	**303**

104. Table 4 below illustrates that Gingerbread and Lollipop's failures of the JCK Signature Test arise from their use of subsets of classes compared to Java SE 6 and 7, respectively. Since subsetting involves the absence of required classes, it will always result in failure of the JCK Signature Test.

Table 4: Summary of manual evaluation of "missing" method errors from the results of the Java SE 7 TCK test against the Android Lollipop runtime implementation

Class Name	Missing TCK Signatures from Lollipop Test	Java SE 7 Signature & Line #
java.util.Calendar	method public boolean java.util.Calendar.isWeekDateSupported()	Calendar.java #2224: *public boolean isWeekDateSupported()*
java.security.Permission	method public abstract boolean java.security.Permission.equals(java.lang.Object)	Security.java #135: *public abstract boolean equals(Object obj)*
java.net.URLConnection	method public long java.net.URLConnection.getContentLengthLong()	URLConnection.java #510: *public long getContentLengthLong()*
java.awt.font.NumericShaper	method public final java.util.Set<java.awt.font.NumericShaper$Range> java.awt.font.NumericShaper.getRangeSet()	NumericShaper.java #1216: *public Set<Range> getRangeSet()*
java.lang.ClassLoader	method protected static boolean java.lang.ClassLoader.registerAsParallelCapable()	ClassLoader.java #1243: *protected static boolean registerAsParallelCapable()*
java.io.File	method public java.nio.file.Path java.io.File.toPath()	File.java #2187: *public Path toPath()*
java.text.DecimalFormatSymbols	method public final static java.text.DecimalFormatSymbols java.text.DecimalFormatSymbols.getInstance()	DecimalFormatSymbols.java #127: *public static final DecimalFormatSymbols getInstance()*
java.util.regex.Matcher	method public final java.lang.String java.util.regex.Matcher.group(java.lang.String)	Matcher.java #520: *public String group(String name)*
java.util.logging.Logger	method public final static java.util.logging.Logger java.util.logging.Logger.getGlobal()	Logger.java #215: *public static final Logger getGlobal()*
java.sql.Connection	method public abstract int java.sql.Connection.getNetworkTimeout() throws java.sql.SQLException	Connection.java #1485: *int getNetworkTimeout() throws SQLException*

105. Likewise, Table 5 below illustrates that Gingerbread and Lollipop's failures of the JCK Signature Test also stem from their use of supersets of classes compared to Java SE 6 and 7, respectively. Since supersetting involves the addition of superfluous classes, it will always result in failure of the JCK Signature Test.

Table 5: Summary of manual verification of "added" errors from the results of the Java SE 7 TCK test against the Android Lollipop runtime implementation

Class Name	Added TCK Signatures from Lollipop Test	Android Lollipop Signature & Line #
java.beans.PropertyChangeListenerProxy	method public java.util.EventListener java.util.EventListenerProxy.getListener()	PropertyChangeListenerProxy.java #56: *PropertyChangeListener listener = (PropertyChangeListener) getListener()*
java.util.concurrent.ForkJoinPool	method public boolean java.util.concurrent.ForkJoinPool.awaitQuiescence(long,java.util.concurrent.TimeUnit)	ForkJoinPool.java #3023: *public boolean awaitQuiescence(long timeout, TimeUnit unit)*
java.util.concurrent.TimeUnit	method public final long java.util.concurrent.TimeUnit.convert(long,java.util.concurrent.TimeUnit)	TimeUnit.java #166: *public long convert(long sourceDuration, TimeUnit sourceUnit)*
javax.xml.validation.SchemaFactory	method public static javax.xml.validation.SchemaFactory javax.xml.validation.SchemaFactory.newInstance(java.lang.String)	SchemaFactory.java #180: *public static SchemaFactory newInstance(String schemaLanguage)*
java.util.ResourceBundle	method public static java.util.ResourceBundle java.util.ResourceBundle.getBundle(java.lang.String)	ResourceBundle.java #134: *public static ResourceBundle getBundle(String bundleName) throws MissingResourceException*
java.util.logging.Logger	method public static java.util.logging.Logger java.util.logging.Logger.getGlobal()	Logger.java #392: *public static Logger getGlobal()*
java.util.ResourceBundle$Control	method public static java.util.ResourceBundle$Control java.util.ResourceBundle$Control.getControl (java.util.List<java.lang.String>)	ResourceBundle.java #763: *public static Control getControl(List<String> formats)*

106. In conclusion, the Android platform's implementation of the 37 Java APIs fails the JCK Signature Test because it represents both a subset and superset of the Java SE API Specification. This failure establishes that Google's Android platform implementations deviate from the compatibility standards imposed by Java's testing suites.

B. Android's Runtime Environment and Virtual Machine are incompatible

107. A runtime environment broadly refers to the set of tools and resources that are used to execute applications that are created for a software platform. The Android and Java SE platforms

have differing runtime environments in two important ways: (1) the respective virtual machines used by each are different, and (2) the standard bytecode format that is produced at the compile-time step and executed by these virtual machines is different. These differences are critical to the loss of compatibility in that byte code from Android will not run on the Java Virtual Machine and Java byte code will not run on the Android Virtual Machine.

108. The Java SE platform uses the Java Virtual Machine ("JVM").[37] The Android platform originally used the Dalvik virtual machine, which has been replaced by the ART (Android Runtime) virtual machine in recent versions of Android.[38]

1) Overview of Java and Android Virtual Machines

109. The Java and Android virtual machines are similar in that they accept as input a form of bytecode that is compiled ahead of time from Java source files. The standard bytecode format used by the JVM is class bytecode, with the .class file extension. The Java Language Specification provides the standards by which class bytecode is created and used by the Java SE platform.[39] The Dalvik VM and ART accept dex bytecode, which carries the .dex file extension.[40]

110. When Android applications are compiled, the constituent Java source files are first compiled into class bytecode of the same format used by the JVM. However, the Android compilation step introduces an additional post-processing tool called "dx" that transforms class bytecode into the dex bytecode used by Android's VMs to execute apps. This process means Android introduces an extra step of compilation that takes bytecode that is JVM-readable and intentionally transforms it into the dex bytecode format.

2) Dex Bytecode cannot be executed by the JVM

[37] *See* Tim Lindholm, Frank Yellin, Gilad Bracha & Alex Buckley, *The Java Virtual Machine Specification*, Oracle (Feb. 13, 2015), https://docs.oracle.com/javase/specs/jvms/se8/jvms8.pdf (for the most recent edition of the Java Virtual Machine specification, which describes in intricate detail the structure and bytecode format of the JVM).

[38] *See* Art and Dalvik, Android, https://source.android.com/devices/tech/dalvik (last visited January 7, 2016).

[39] James Gosling, Bill Joy, Guy Steele, Gilad Bracha & Alex Buckley, *The Java Language Specification,* Oracle (Feb 13, 2015), https://docs.oracle.com/javase/specs/jls/se8/jls8.pdf.

[40] *See* Dalvik Executable Format, Android, https://source.android.com/devices/tech/dalvik/dex-format.html (last visited January 7, 2016).

111. The dex bytecode format that is the standard for the Android runtime environment is not understood by the JVM, and applications that are compiled into dex bytecode cannot be executed by the JVM. The following experiment was conducted using the JVM from Java SE 5.

112. To illustrate this incompatibility, a sample application was created, and transformed into class bytecode using the Java compiler. To ensure that any errors did not stem from the application itself, the class bytecode was inputted into the JVM to verify that it would run successfully. From here, the dx tool was used to transform the class bytecode into dex bytecode. Then, the outputted .dex file was re-named to a .class file.[41]

113. When attempting to execute the test application bytecode on the JVM, it failed and produced an error relating to the Java Native Interface, or JNI.[42] This error implies that the input file violates the binary compatibility standard expected by the JNI, which in turn shows that the JVM is not compatible with dex bytecode.

3) Class Bytecode cannot be executed by the Dalvik VM

114. The class bytecode format that is the standard for the JVM cannot be interpreted by the Dalvik VM. The following test was conducted using an emulated device running Android Gingerbread, which carries version 1.4.0 of the Dalvik VM.

115. Dalvik 1.4.0 only accepts bytecode that is archived into .jar files. As a result, this test involved compiling a test application into class bytecode using the Java compiler, then using the jar tool provided in the JDK to package this bytecode into a .jar file. The contents of the .jar file were then altered so that they would appear to the VM to be a .jar file of dex bytecode.[43]

[41] The file extension is renamed here so that the JVM actually attempts to execute the bytecode contents of the file. When submitting a .dex file as input, the JVM will return an error simply because the .dex file extension is not recognized by the JVM. Renaming the extension forces the JVM to interpret the dex bytecode inside the file, which is an appropriate test of whether or not the VM can actually read dex bytecode.

[42] The JNI provides an interface between high-level Java source code and lower-level code written in C or C++ (native code), and for embedding the JVM into native applications. In short, the JNI is an interface that ensures binary compatibility of native method libraries for any JVM implementation.

[43] This restructuring is done for the same reason that the file extension is renamed in the previous test - to 'trick' the VM into interpreting the contents such that the error message thrown is a faithful representation of its inability to interpret the bytecode, rather than a contrived error that results from a failure to recognize a file extension or filename which pre-empts the bytecode interpretation step.

116. When the jar file was supplied to the Dalvik VM, an error was returned indicating that the magic number of the jar file was different from what was expected by the VM. The magic number of a file is a unique identifier found in the contents of a file that indicates the bytecode format. This error indicates that the VM fails to interpret the bytecode because it cannot identify the magic number found in the bytecode instructions. This shows that the Dalvik VM is not compatible with class bytecode.

X. CONCLUSIONS

117. Since its inception in the mid 1990s, the Java platform has fostered a robust, dynamic community of both application and infrastructure developers. Java API packages offer the cogent, portable set of tools that enable developers to easily create programs using the Java platform. Compatibility is an essential virtue of the Java platform, as indicated by the "Write Once, Run Anywhere" slogan created by Sun Microsystems to emphasize Java's cross-platform benefits with respect to portability and compatibility. Both Sun and Oracle have always actively sought to preserve compatibility, both to protect their commercial investments in Java and for the greater benefit of developers and end-users.

118. Google's copying of the 37 Java APIs is a reflection of the value of the Java API packages' key attributes concerning productivity (especially leveraging the large group of developers that are familiar with the Java API packages).

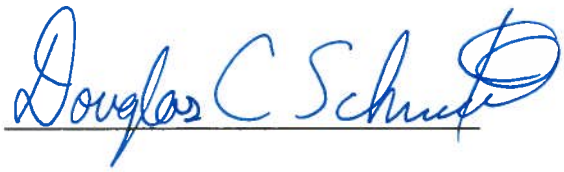
119. The Android platform violates the fundamental "Write Once, Run Anywhere" paradigm of the Java ecosystem, and is thus incompatible with the Java platform. Android fails to demonstrate compatibility with Java when subjected to various forms of empirical testing, including Oracle's industry benchmark Technology Compliance Kit as well as independent tests and measurements because it represents both a subset and superset of the Java SE API Specification. Moreover, Android's incompatibility manifests in both its software development kits and its virtual machine runtime environments.

120. The Android platform remains persistently incompatible with Java. I have seen no evidence that Google has made any attempts to ensure Android's compatibility with the broader Java ecosystem and the TCK.

121. The Android platform is critically dependent on the 37 Java APIs, individually and collectively. I find this critical dependence in the fact that the build process used to create the code that runs Android on mobile devices fails to generate the necessary executable code (called executable image files) in the absence of the 37 Java API packages at issue. The build process fails if the files from even one of the 37 API packages at issue is removed. In fact, the build process also fails when I remove only the copied lines of declaring code while leaving the rest of the files in place. As a result, Android will not run on a mobile device without all of the copied declaring code and the full set of files for the 37 API packages at issue.

XI. ATTESTATIONS

123. I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct. Executed on this 8th day of January, 2016, in San Francisco, California.



PROOF OF SERVICE BY KITEWORKS

I, José E. Valdés, am over the age of eighteen years old and not a party to the within-entitled action. My place of employment and business address is Orrick, Herrington & Sutcliffe LLP, 1000 Marsh Road, Menlo Park, California 94025.

On January 8, 2016, I served the following documents:

EXPERT REPORT OF PROFESSOR DOUGLAS C. SCHMIDT, Ph.D.

on the interested parties in this action by electronic service [Fed. Rule Civ. Proc. 5(b)] by electronically mailing a true and correct copy, pursuant to the parties agreement, to the following email addresses:

DALVIK-KVN@kvn.com
JCooper@fbm.com
gglas@fbm.com

I declare under penalty of perjury under the laws of the State of California and the United States that the foregoing is true and correct.

Executed on January 8, 2016, at San Francisco, California.

/s/ José E. Valdés
José E. Valdés

I. INDEX OF APPENDICES

Appendix	Title
A	Curricula Vitae of Douglas C. Schmidt
B	Materials Considered
C	The Java Class Libraries
D	Evolution of Java APIs
E	Glossary of Terms
F	Java SE 5.0 Platform
G	Packages in Java SE5 and Copied API Packages
H	Android Platform Java SE 5.0 Platform
I	List of Copied Oracle APIs
J	Android API Levels and Versions
K	Error Log from Android Build Test No. 1 Romoval of the Entire Set of 37 APIs
L	Error Log from Android Build Test No. 2: Removal of Individual API Source Files
M	Error Log from Android Build Test No. 3A: Removal of Source File Declaring Codes from java.text.Annotation
N	Error Log from Android Build Test No. 3B: Removal of all copied declaring codes from all 37 APIs

II. APPENDIX A: CURRICULA VITAE OF DOUGLAS C. SCHMIDT

Dr. Douglas Craig Schmidt

Educational Background

- **Ph.D. Computer Science**, summer 1994, University of California, Irvine

  Dissertation: "An Object-Oriented Framework for Experimenting with Alternative Process Architectures for Parallelizing Communication Subsystems."

  Co-advisors: Dr. Tatsuya Suda and Dr. Richard W. Selby.

- **M.S. Computer Science**, summer 1990, University of California, Irvine, specializing in software engineering.
- **M.A. Sociology**, summer 1986, College of William and Mary, Williamsburg, Virginia

  Thesis: "A Statistical Analysis of University Resource Allocation Policies."

  Advisor: Dr. Michael A. Faia.

- **B.A. Sociology**, summer 1984, College of William and Mary, Williamsburg, Virginia.

Professional Experience

1. **1/03 – present: Full Professor with tenure**

   Conducting research on patterns, optimizations, and experimental analysis of advanced generative software techniques that facilitate the development of distributed real-time and embedded middleware and model driven architectures running over high-speed networks and interconnects in the Department of Electrical Engineering and Computer Science at Vanderbilt University.

2. **12/04 – 06/14: Associate Chair of Computer Science and Engineering**

   Provided intellectual leadership within the CS program. Coordinate with EECS Chair to assist in CS and CompE (CS&E) curriculum development and course staffing. Assist the faculty in building industry and federal programs centered in CS&E and IT for EECS. Assist the Chair in mentoring young CS&E faculty. Assist the EECS Chair in improving the ranking of the CS&E programs. Assist the Chair in increasing the quality and number of undergraduate and graduate student applications to the CS&E programs.

3. **4/13 – present: Member of the Board of Directors at Real-Time Innovations (RTI)**.

   Work with the CEO and other members of the Board of Directors of RTI to help assess company technical and business strategy.

4. **1/12 – present: Visiting Scientist at the Software Engineering Institute**

   Assist the SEI Director's Office in formulating the SEI's technology strategy for R&D projects and external relationships by aligning the expertise of the SEI technical staff to identify and respond to the needs of sponsors, customers, and partners and help the SEI shape future innovations in complex software-reliant systems.

5. **7/11 – 7/13: Adjunct Professor of Software Engineering** in the Institute for Software Research in the School of Computer Science at Carnegie Mellon University.

6. **9/10 – 12/11: Deputy Director and Chief Technology Officer at the Software Engineering Institute (SEI)**

   Leading the formulation of the SEI's technology strategy for R&D projects and external relationships by aligning the expertise of the SEI technical staff to identify and respond to the needs of sponsors, customers, and partners and help the SEI shape future innovations in complex software-reliant systems.

7. **07/05 – 8/10: Visiting Scientist at the Software Engineering Institute**

   Assisted Linda Northrop and the Ultra-Large-Scale (ULS) Systems team to define the challenge problems, promising technology areas, and research roadmaps for the national R&D effort on building the software-reliant systems of the future that are likely to have billions of lines of code. This activity is defining a broad, multi-disciplinary research agenda for developing ULS systems of the future.

8. **06/09 – 8/10: Chief Technology Officer for Zircon Computing**

   Assisted in the strategic direction of Zircon Computing technology development in the areas of adaptive distributed computing middleware for high-performance and real-time applications. Help to formulate the technology strategy for open-source middleware platforms, R&D partnerships, and external relationships.

9. **10/06 – 5/09: Chief Technology Officer for PrismTechnologies**

   Assisted in the strategic direction of PrismTechnologies technology development in the areas of open-source middleware platforms and model-driven tools. Help to formulate the technology strategy for open-source middleware platforms and model-driven tools, R&D partnerships, and external relationships.

10. **6/07 – 8/07: Visiting Professor at Trinity College Dublin**

    Worked with Professor Vinny Cahill and the Distributed Systems Group at Trinity College on topics pertaining to service-oriented architectures and autonomic computing.

11. **3/02 – 12/02: Program Manager**

    Led the national effort on middleware as a Program Manager for over $60 million dollars of funding at the DARPA Information Exploitation Office (IXO). Programs

include Program Composition for Embedded Systems (PCES) and National Experimentation Platform for Hybrid and Embedded Systems (NEPHEST).

12. **9/01 – 3/02: Deputy Director**

    Served as the Deputy Director for the DARPA Information Technology Office (ITO), helping to set the national IT research and development agenda and manage programs on autonomous systems, network-centric command and control systems, distributed real-time and embedded systems, and augmented cognition for the U.S. Department of Defense.

13. **6/00 – 3/02: Program Manager**

    Led the national effort on middleware as a Program Manager for over $60 million dollars of funding at the DARPA Information Technology Office (ITO). Programs included the Program Composition for Embedded Systems (PCES).

14. **6/01 – 6/02: Co-chair for the Software Design and Productivity (SDP) Coordinating Group**

    The SDP Coordinating Group formulates the multi-agency research agenda in fundamental software design for the Federal government's Information Technology Research and Development (IT R&D) Program, which is the collaborative IT research effort of the major Federal science and technology agencies.

15. **8/99 – 2002: Associate Professor with tenure**

    Conducted research on patterns, implementation, and experimental analysis of object-oriented techniques that facilitate the development of high-performance, real-time distributed object computing systems on parallel processing platforms running over high-speed networks and embedded system interconnects in the Department of Computer Engineering at the University of California, Irvine.

16. **6/99 – 8/99: Associate Professor with tenure**

    Conducted research on patterns, implementation, and experimental analysis of object-oriented techniques that facilitate the development of high-performance, real-time distributed object computing systems on parallel processing platforms running over high-speed networks and embedded system interconnects in the Department of Computer Science and the Department of Radiology at Washington University in St. Louis.

17. **6/98 – 6/99: Associate Professor without tenure (early promotion)**

    Conducted research on patterns, implementation, and experimental analysis of object-oriented techniques that facilitate the development of high-performance, real-time distributed object computing systems on parallel processing platforms running over high-speed networks and embedded system interconnects in the Department of Computer Science and the Department of Radiology at Washington University in St. Louis.

18. **8/94 – 6/98: Assistant Professor**

    Conducted research on object-oriented patterns and techniques for developing highly extensible, high-performance communication frameworks in the Department of Computer Science and the Department of Radiology at Washington University in St. Louis.

19. **3/91 – 8/94: Computer System Design Research Assistant**

    Developed an object-oriented framework for multi-processor-based communication subsystems with Professor Tatsuya Suda at the University of California, Irvine.

20. **6/90 – 11/90: Member of the Technical Staff**

    Worked as a software engineer for Independence Technologies, which was one of the largest suppliers of enterprise-level TUXEDO systems, providers of professional services, and developers of management and connectivity software to support OLTP environments.

21. **8/88 – 3/91: Software Engineering Research Assistant**

    Devised measurement-guided software development techniques for large-scale software systems with Professor Richard Selby at the University of California, Irvine.

22. **6/88 – 8/88: Research Assistant**

    Studied the impact of computing on end-users in forty U.S. city governments with Dr. John King and the URBIS project at the Public Policy Research Organization, University of California, Irvine.

23. **1/85 – 8/86: Sociology Research Assistant**
    Examined university resource allocation policies via statistical analysis with Dr. Michael Faia at the College of William and Mary, Williamsburg, Virginia.

Publications

In Print

- **Refereed Journal Publications**

  J112 Nick Guertin, Brian Womble, Paul Bruhns, Douglas C. Schmidt, Adam Porter, and Bill Antypas, "Management Strategies for Software Infrastructure in Large-Scale Cyber-Physical Systems for the US Navy," Cutter IT Journal, Vol. 28, No. 5, May 2015.

  J111 Jules White, David Benavides, Tripti Saxena, Brian Dougherty, Douglas C. Schmidt, and Jose A. Galindo, "Automated Reasoning for Multi-step Feature Model Configuration Problems, the Journal of Systems and Software, Elsevier, 2015 (to appear).

  J110 Jules White, Josi A. Galindo, Tripti Saxena, Brian Dougherty, David Benavides, Douglas C. Schmidt, "Evolving Feature Model Configurations in Software Product Lines," Journal of Systems and Software, Volume 87, 2014, pp. 119-136.

J109 Akram Hakiri, Aniruddha S. Gokhale, Pascal Berthou, Douglas C. Schmidt, Thierry Gayraud, Software-Defined Networking: Challenges and Research Opportunities for the Future Internet," Journal of Computer Networks, Volume 75, 2014, pp. 453-471.

J108 Hamilton Turner, Brian Dougherty, Jules White, Jonathan Preston, Russell Kegley, Douglas C. Schmidt, and Aniruddha Gokhale, "DRE System Performance Optimization with the SMACK Cache Efficiency Metric," Elsevier Journal of Systems and Software, Volume 98, 2014, pp. 25-43.

J107 Akram Hakiri, Pascal Berthoua, Aniruddha Gokhale, Douglas C. Schmidt, Gayraud Thierry, "Supporting SIP-based Data Distribution Service End-to-End QoS in WANs," the Elsevier Journal of Systems and Software, Volume 95, September 2014, pp. 100-121.

J106 Jules White, Douglas C. Schmidt, and Mani Golparvar-Fard, "Applications of Augmented Reality," IEEE Proceedings Special issue on Applications of Augmented Reality, Vol 102, No. 2., February 2014, pp. 120-123.

J105 Nickolas H. Guertin, Paul Bruhns, Douglas C. Schmidt, and Adam Porter, "Experiences Using Online War Games to Improve the Business of Naval Systems Acquisition," Cutter Journal of Information Technology Management, Vol. 27, No. 5, May 2014, pp 13-18.

J104 Michael McLendon, Bill Scherlis, and Douglas C. Schmidt, "Addressing Software Sustainment Challenges for the DoD," STSC CrossTalk, The Journal of Defense Software Engineering special issue on Legacy Systems Software, January, volume 27, number 1, 2014, pp. 27-32.

J103 Akram Hakiri, Pascal Berthoua, Aniruddha Gokhale, Douglas C. Schmidt, Gayraud Thierry, "Supporting End-to-end Scalability and Real-time Event Dissemination in the OMG Data Distribution Service over Wide Area Networks," Elsevier Journal of Systems and Software, volume 86, number 10, October, 2013, pp. 2574–2593.

J102 William Otte, Aniruddha Gokhale, and Douglas C. Schmidt, "Efficient and Deterministic Application Deployment in Component-based, Enterprise Distributed, Real-time, and Embedded Systems," Elsevier Journal of Information and Software Technology, Vol. 55, No. 2, Feb 2013, 475–488.

J101 Dr. Douglas Schmidt, Anita Carleton, Erin Harper, Mary Ann Lapham, Ipek Ozkaya, and Linda Parker Gates, 'What Will It Take to Achieve Agility-at-Scale? ", Cutter IT Journal, edited by Hillel Glazer, November 2012, pp. 34-39.

J100 Brian Dougherty, Jules White, and Douglas C. Schmidt, "Model-driven Auto-scaling of Green Cloud Computing Infrastructure," the Elsevier International Journal of Future Generation Computing Systems, Special Issue on Green Computing Systems, Volume 28, Number 2, February, 2012 Pages 371-378.

J99 Joe Hoffert, Douglas C. Schmidt, and Aniruddha Gokhale, "Evaluating Timeliness and Accuracy Trade-offs of Supervised Machine Learning for Adapting Enterprise DRE Systems in Dynamic Environments," the

International Journal of Computational Intelligence Systems, Volume 4, Number 5, September-October 2011, pp. 806-816.

J98 James Hill, Pooja Varshneya, and Douglas C. Schmidt, "Evaluating Distributed Real-time and Embedded System Test Correctness using System Execution Traces," Central European Journal of Computer Science, Volume 1, Number 2, August 2011, pp. 167-184.

J97 Brian Dougherty, Jules White, and Douglas C. Schmidt, "Automated Software and Hardware Evolution Analysis for Distributed Real-time and Embedded Systems," the Central European Journal of Computer Science, Volume 1, Number 1, July 2011, pp. 36-57.

J96 James Hill, Hunt Sutherland, Paul Stodinger, Thomas Silveria, Douglas C. Schmidt, John Slaby, and Nikita Visnevski, "OASIS: An Architecture for Dynamic Instrumentation of Enterprise Distributed Real-time and Embedded Systems," the International Journal of Computer Systems Science and Engineering, Special Issue on Real-time Systems, Volume 26, Number 6, November 2011, pp. 413-430.

J95 Jules White, Brian Dougherty, Chris Thompson, Douglas C. Schmidt, "ScatterD: Spatial Deployment Optimization with Hybrid Heuristic/Evolutionary Algorithms," ACM Transactions on Autonomous and Adaptive Systems Special Issue on Spatial Computing, Volume 6 Issue 3, September 2011, 18:1–18:25.

J94 Jules White, Chris Thompson, Hamilton Turner, Brian Dougherty, and Douglas C. Schmidt, WreckWatch: Automatic Traffic Accident Detection and Notification with Smartphones, Journal of Mobile Networks and Applications, Volume 16 Issue 3, July 2011, Pages 285-303.

J93 Jules White, Brian Dougherty, Richard Schantz, Douglas C. Schmidt, Adam Porter, and Angelo Corsaro, "R&D Challenges and Solutions for Highly Complex Distributed Systems: a Middleware Perspective," the Springer Journal of Internet Services and Applications special issue on the Future of Middleware, Volume 2, Number 3, December 2011, pp. 1–8.

J92 Joe Hoffert, Aniruddha Gokhale, and Douglas C. Schmidt, "Autonomic Adaptation of Publish/Subscribe Middleware in Dynamic Environments," the International Journal of Adaptive, Resilient and Autonomic Systems (IJARAS), 2(4), 1-24, October-December 2011, pp. 1–24.

J91 Joe Loyall, Matt Gillen, Aaron Paulos, Larry Bunch, Marco Carvalho, James Edmondson, Douglas C. Schmidt, Andrew Martignoni, and Asher Sinclair, "Dynamic Policy-Driven Quality of Service in Service-Oriented Information Management Systems," Wiley journal on Software: Practice and Experience, December 2011, volume 41, number 12, pp. 1459–1489.

J90 Michael Stal, Douglas C. Schmidt, and Will Otte, "Efficiently and Transparently Automating Scalable On-demand Activation and Deactivation of Services with the Activator Pattern," Software: Practice and Experience, special issue on Pattern Languages: Addressing Challenges, Edited by

Mohamed Fayad and Shivanshu Singh, volume 41, number 10, October 2011, Wiley and Sons, pp. 1-16.

J89 Brian Dougherty, Jules White, Douglas C. Schmidt, Russell Kegley, and Jonathan Preston, "Deployment Optimization for Embedded Flight Avionics Systems," STSC CrossTalk, The Journal of Defense Software Engineering, November/December, volume 24, number 6, 2011, pp. 1-8.

J88 Brian Dougherty, Daniel Guymon, Douglas C. Schmidt, and Jules White, "Overcoming Cellular Connectivity Limitations with M2Blue Autonomic Distributed Data Caching," Autonomic Computing for Computer Society of India Magazine, CSI Communications, August 2011, pp. 12-15.

J87 Friedhelm Wolf, Jaiganesh Balasubramanian, Sumant Tambe, Aniruddha Gokhale, and Douglas C. Schmidt, Supporting Component-based Failover Units in Middleware for Distributed Real-time and Embedded Systems, the Elsevier Journal of System Architectures (JSA): Embedded Systems Design, Special Issue on Real-time and Embedded Systems, May, 2011 pp. 597-613.

J86 Jules White, David Benavides, Douglas C. Schmidt, Pablo Trinidad, Antonio Ruiz-Cortes, Brian Doughtery, "Automated Diagnosis of Feature Model Configurations," The Journal of Systems and Software, Special Issue on Software Product-lines, Volume 83, Issue 7, July, 2010, pp. 1094-1107.

J85 Jules White, Brian Doughtery, and Douglas C. Schmidt, "ASCENT: An Algorithmic Technique for Designing Hardware and Software in Tandem, IEEE Transactions on Software Engineering Special Issue on Search-based Software Engineering, November/December 2010 (vol. 36 no. 6), pp. 838-851.

J84 Joe Hoffert, Daniel Mack, and Douglas Schmidt, "Integrating Machine Learning Techniques to Adapt Protocols for QoS-enabled Distributed Real-time and Embedded Publish/Subscribe Middleware," International Journal of Network Protocols and Algorithms (NPA): Special Issue on Data Dissemination for Large-scale Complex Critical Infrastructures, Volume 2, Number 3, 2010, pp. 37-69.

J83 James Hill, James Edmondson, Aniruddha Gokhale, and Douglas C. Schmidt, "Tools for Continuously Evaluating Distributed System Qualities," IEEE Software, July/August, 2010, Volume 27, Number 4, pp. 65-71.

J82 James Edmondson and Douglas C. Schmidt, Multi-Agent Distributed Adaptive Resource Allocation (MADARA), International Journal of Communication Networks and Distributed Systems (IJCNDS), Special Issue on: Grid Computing, Edited by Michal Wozniak and Krzysztof Walkowiak, Volume 5, Number 3, 2010, pp. 229-245.

J81 Jules White, Christin Groba, Sibohan Clarke, Brian Dougherty, Chris Thompson, and Douglas C. Schmidt, "R&D Challenges and Solutions for Mobile Cyber-Physical Applications and Supporting Internet Services," the Springer Journal of Internet Services and Applications, Volume 1, Number 1, 2010, pp. 45-56.

J80 Jules White, Jeff Gray, and Douglas C. Schmidt, "Constraint-based Model Weaving," Transactions on Aspect-Oriented Software Development, Special Issue on Aspects and Model Driven Engineering, eds. Robert France and Jean-Marc Jezequel, pp. 153-190, Volume 5560, Number 6, 2009.

J79 Jules White, Harrison Strowd, and Douglas C. Schmidt, "Creating Self-healing Service Compositions with Feature Modeling and Microrebooting," the International Journal of Business Process Integration and Management (IJBPIM), Special issue on Model-Driven Service-Oriented Architectures, Inderscience Publishers, pp. 35-46, Volume 4, Number 1, 2009.

J78 Nishanth Shankaran, John Kinnebrew, Xenofon Koutsoukos, Chenyang Lu, Douglas C. Schmidt, and Gautam Biswas, "An Integrated Planning and Adaptive Resource Management Architecture for Distributed Real-time Embedded Systems," IEEE Transactions on Computers, Special Issue on Autonomic Network Computing, Special Issue on Autonomic Network Computing, volume 58, number 11, pp. 1485-1498, November 2009.

J77 Jules White, Brian Doughtery, and Douglas C. Schmidt, "Selecting Highly Optimal Architectural Feature Sets with Filtered Cartesian Flattening," the Journal of Software and Systems, Special Issue on Design Decisions and Design Rationale in Software Architecture, Volume 82, Issue 8, pp. 1268-1284, August 2009.

J76 Jules White, James, Hill, Sumant Tambe, Jeff Gray, Aniruddha Gokhale, and Douglas C. Schmidt "Improving Domain-specific Language Reuse through Software Product-line Configuration Techniques, IEEE Software Special Issue: Domain-Specific Languages and Modeling, vol. 26, no. 4, pp. 47-53, July/August 2009.

J75 Jules White and Douglas C. Schmidt, "Automating Deployment Planning with an Aspect Weaver," IET Software Journal Special Issue on Domain-specific Aspect Languages, Volume 3, Issue 3 , p. 167-183, June 2009.

J74 Shanshan Jiang, Yuan Xue, and Douglas C. Schmidt, "Minimum Disruption Service Composition and Recovery in Mobile Ad Hoc Networks, Elsevier Computer Networks Journal, Special Issue on Autonomic and Self-Organizing Systems, Volume 53, Issue 10, Pages 1649-1665, 2009.

J73 Nishanth Shankaran, Douglas C. Schmidt, Xenofon D. Koutsoukos, Yingming Chen, and Chenyang Lu, "Design and Performance Evaluation of an Adaptive Resource Management Framework for Distributed Real-time and Embedded Systems," EURASIP Journal on Embedded Systems (EURASIP JES): Special issue on Operating System Support for Embedded Real-Time Applications, Edited by Alfons Crespo, Ismael Ripoll, Michael Gonzalez Harbour, and Giuseppe Lipari, 2008, Pgs. 47-66.

J72 Aniruddha Gokhale, Krishnakumar Balasubramanian, Jaiganesh Balasubramanian, Arvind Krishna, and George T. Edwards, Gan Deng, Emre Turkay, Jeffrey Parsons, and Douglas C. Schmidt, Model Driven Middleware: A New Paradigm for Deploying and Provisioning Distributed Real-time and Embedded Applications, Elsevier Journal of Science of Computer

Programming: Special Issue on Foundations and Applications of Model Driven Architecture (MDA), Edited by Mehmet Aksit, Volume 73, Issue 1, 1 September 2008, Pgs. 39-58.

J71 Nishanth Shankaran, Xenofon Koutsoukos, Chenyang Lu, Douglas C. Schmidt, and Yuan Xue, "Hierarchical Control of Multiple Resources in Distributed Real-time and Embedded Systems," the Springer Real-time Systems Journal, Volume 39, Numbers 1-3, August, 2008, pgs. 237-282.

J70 Douglas C. Schmidt, Angelo Corsaro, and Hans Van'T Hag, "Addressing the Challenges of Tactical Information Management in Net-Centric Systems with DDS," CrossTalk special issue on Distributed Software Development, pgs. 24-29, May 2008.

J69 Jules White, Douglas C. Schmidt, Egon Wuchner, and Andrey Nechypurenko, "Automatically Composing Reusable Software Components for Mobile Devices," Journal of the Brazilian Computer Society (JBCS), Special Issue in Software Reuse: Methods, Processes, Tools and Experiences, Sociedade Brasileira de Computacao, Porto Alegre, Volume 14, Number 1, pgs. 25-44, March, 2008.

J68 Jules White, Douglas C. Schmidt, Andrey Nechypurenko, and Egon Wuchner, "Model Intelligence: an Approach to Modeling Guidance," UPGRADE Journal, Volume 9, Number 2, pgs. 22-28, April 2008.

J67 Douglas C. Schmidt, "Beyond Objects: Evaluating Technologies for Developing Distributed Systems," Computer Society of India Communications, Special Issue on OO Technologies, edited by Debasish Jana, pgs. 30-37, February 2008.

J66 Jules White, Douglas C. Schmidt, and Aniruddha Gokhale, "Simplifying Autonomic Enterprise Java Bean Applications via Model-driven Engineering and Simulation," The Journal of Software and System Modeling, Volume 7, Number 1, February, 2008, pgs. 3-23.

J65 Cemal Yilmaz, Adam Porter, Arvind S. Krishna, Atif Memon, Douglas C. Schmidt, and Aniruddha Gokhale, "Reliable Effects Screening: A Distributed Continuous Quality Assurance Process for Monitoring Performance Degradation in Evolving Software Systems," IEEE Transactions on Software Engineering Vol. 33, No. 8, August 2007, pgs. 510-525.

J64 Frank Buschmann, Kevlin Henning, and Douglas C. Schmidt, "Past, Present, and Future Trends in Software Patterns," IEEE Software special issue on Patterns, Vol. 24, No. 4, July/August, 2007, pgs. 31-37.

J63 Patrick Lardieri, Jaiganesh Balasubramanian, Douglas C. Schmidt, Gautam Thaker, Aniruddha Gokhale, and Tom Damiano, A Multi-layered Resource Management Framework for Dynamic Resource Management in Enterprise DRE Systems, the Journal of Systems and Software: special issue on Dynamic Resource Management in Distributed Real-Time Systems, editors C. Cavanaugh and F. Drews and L. Welch, Vol 80, Issue 7, July 2007, pgs. 984-996.

J62 Janos Sztipanovits, John Bay, Larry Rohrbough, Shankar Sastry, Douglas C. Schmidt, Don Wilson, and Don Winters, "Escher: A New Technology Transitioning Model," IEEE Computer, Vol. 40, No. 3, March 2007, pgs. 90-92.

J61 Venkita Subramonian, Gan Deng, Christopher Gill, Jaiganesh Balasubramanian, Liang-Jui Shen, William Otte, Douglas C. Schmidt, Aniruddha Gokhale, and Nanbor Wang, "The Design and Performance of Component Middleware for QoS-enabled Deployment and Configuration of DRE Systems," Elsevier Journal of Systems and Software, Special Issue Component-Based Software Engineering of Trustworthy Embedded Systems, pp. 668–677, volume 80, number 5, March, 2007.

J60 Krishnakumar Balasubramanian, Jaiganesh Balasubramanian, Jeff Parsons, Aniruddha Gokhale, and Douglas C. Schmidt, "A Platform-Independent Component Modeling Language for Distributed Real-time and Embedded Systems," Elsevier Journal of Computer and System Sciences, Volume 73, Number 2, March 2007, pgs. 171 - 185.

J59 Adam Porter, Atif Memon, Cemal Yilmaz, Douglas C. Schmidt, and Bala Natarajan, "Skoll: A Process and Infrastructure for Distributed Continuous Quality Assurance, IEEE Transactions on Software Engineering, 2007, Vol. 33, No. 2, February 2007, pgs. 124-141.

J58 Richard E. Schantz, Douglas C. Schmidt, Joseph P. Loyall, and Craig Rodrigues, "Controlling Quality-of-Service in Distributed Real-time and Embedded Systems via Adaptive Middleware," the Wiley Software Practice and Experience journal special issue on Experiences with Auto-adaptive and Reconfigurable Systems, co-editors Mehmet Aksit, Zied Choukair, and Tzilla Elrad, vol. 36, no. 11-12, September 2006, pgs. 1189 - 1208.

J57 Douglas C. Schmidt, "Model-Driven Engineering, IEEE Computer, Vol. 39. No. 2, February 2006, pp. 41–47.

J56 Arvind S. Krishna, Aniruddha Gokhale, Douglas C. Schmidt, John Hatcliff, and Venkatesh Prasad Ranganat, "Towards Highly Optimized Real-time Middleware for Software Product-line Architectures," ACM SIGBED Review, Volume 3, No. 1, January 2006, pgs. 12-16.

J55 Gabor Madl, Sherif Abdelwahed, and Douglas C. Schmidt, "Verifying Distributed Real-time Properties of Embedded Systems via Graph Transformations and Model Checking, Real-time Systems Journal, vol 33, no. 1-3, pgs. 77-100, July 2006.

J54 Cemal Yilmaz, Adam Porter, Atif Memon, Arvind S. Krishna, Douglas C. Schmidt, and Aniruddha Gokhale, Techniques and Processes for Improving the Quality and Performance of Open-Source Software, Software Process - Improvement and Practice Journal: Special Issue on Free/Open Source Software Processes, vol 11, no 2, May 2006, pgs. 163-176.

J53 Krishnakumar Balasubramanian, Arvind S. Krishna, Emre Turkay, Jaiganesh Balasubramanian, Jeff Parsons, Aniruddha Gokhale, and Douglas C. Schmidt, "Applying Model-Driven Development to Distributed Real-time and

Embedded Avionics Systems, the International Journal of Embedded Systems, special issue on Design and Verification of Real-Time Embedded Software, April 2005.

J52 Arvind S. Krishna, Cemal Yilmaz, Adam Porter, Atif Memon, Douglas C. Schmidt, and Aniruddha Gokhale, "Distributed Continuous Quality Assurance Process for Evaluating QoS of Performance Intensive Software," Studia Informatica Universalis, Volume 4, No. 1, pp. 53-72, March 2005.

J51 Janos Sztipanovits, Gautam Biswas, Ken Frampton, Andy Gokhale, Larry Howard, Gabor Karsai, John Koo, Xenofon Koutsoukos, and Douglas C. Schmidt, "Introducing Embedded Software and Systems Education and Advanced Learning Technology in an Engineering Curriculum," ACM Transactions in Embedded Computing Systems - Special Issue on Education, edited by Alan Burns and Alberto Sangiovanni-Vincentelli, Vol 4, No. 1, pp. 549-568, August 2005.

J50 Arvind S. Krishna, Nanbor Wang, Balachandran Natarajan, Aniruddha Gokhale, Douglas C. Schmidt and Gautam Thaker, "CCMPerf: A Benchmarking Tool for CORBA Component Model Implementations", The International Journal of Time-Critical Computing Systems, Springer, Vol. 29, No. 2-3, pp. 281-308, March-April 2005.

J49 Chris Gill, Jeanna M. Gossett, David Corman, Joseph P. Loyall, Richard E. Schantz, Michael Atighetchi, and Douglas C. Schmidt, "Integrated Adaptive QoS Management in Middleware: An Empirical Case Study," The International Journal of Time-Critical Computing Systems, Springer, Vol. 29, Nos. 2-3, pp. 101-130, March-April 2005.

J48 Aniruddha Gokhale, Balachandran Natarajan, Douglas C. Schmidt, and Joseph Cross, "Towards Real-time Fault-Tolerant CORBA Middleware," Cluster Computing: the Journal on Networks, Software, and Applications Special Issue on Dependable Distributed Systems, edited by Alan George, Volume 7, Number 4, October 2004.

J47 Arvind S. Krishna, Cemal Yilmaz, Atif Memon, Adam Porter, Douglas C. Schmidt, Aniruddha Gokhale, and Balachandran Natarajan, "Preserving Distributed Systems Critical Properties: a Model-Driven Approach," the IEEE Software special issue on the Persistent Software Attributes, Nov/Dec 2004.

J46 Christopher Gill, Douglas C. Schmidt, Yamuna Krishnamurthy, Irfan Pyarali, Louis Mgeta, Yuanfang Zhang, and Stephen Torri, "Enhancing the Adaptivity of Distributed Real-time and Embedded Systems via Standard QoS-enabled Dynamic Scheduling Middleware," the Journal of the Brazilian Computer Society (JCBS) special issue on Adaptive Software Systems, Volume 10, Number 1, pp. 19-30, 2004.

J45 Douglas C. Schmidt, Aniruddha Gokhale, and Balachandran Natarajan, "Frameworks: Why They Are Important and How to Apply Them Effectively," ACM Queue magazine, Volume 2, Number 5, July/August 2004.

J44 Douglas C. Schmidt, Richard Schantz, Aniruddha Gokhale, and Joe Loyall, "Middleware R&D Challenges for Distributed Real-time and Embedded Systems," ACM SIGBED Review, Volume 1, No. 1, April 2004.

J43 Angelo Corsaro and Douglas C. Schmidt, "The Design and Performance of Real-time Java Middleware," Special Issue on Middleware for the IEEE Transactions on Parallel and Distributed Systems, guest editor Rachid Guerraoui, Volume 14, Number 11, November 2003.

J42 Irfan Pyarali, Douglas C. Schmidt, and Ron Cytron, "Techniques for Enhancing Real-time CORBA Quality of Service," the IEEE Proceedings Special Issue on Real-time Systems, co-editors Yann-Hang Lee and C. M. Krishna, Volume 91, Number 7, July 2003.

J41 Nanbor Wang, Douglas C. Schmidt, Aniruddha Gokhale, Christopher D. Gill, Balachandran Natarajan, Craig Rodrigues, Joseph P. Loyall, and Richard E. Schantz, "Total Quality of Service Provisioning in Middleware and Applications," Elsevier Journal of Microprocessors and Microsystems, Volume 26, Number 9-10, January 2003.

J40 Christopher D. Gill, Douglas C. Schmidt, and Ron Cytron, "Multi-Paradigm Scheduling for Distributed Real-Time Embedded Computing," IEEE Proceedings Special Issue on Modeling and Design of Embedded Systems, Volume 91, Number 1, January, 2003.

J39 Aniruddha Gokhale, Bala Natarajan, Douglas C. Schmidt, and Nanbor Wang, "Modeling and Synthesis of Middleware Components," Communications of the ACM, special issue on Enterprise Components, Services and Business Rules, edited by Ali Arsanjani, October 2002.

J38 Douglas C. Schmidt and Carlos O'Ryan, "Patterns and Performance of Distributed Real-time and Embedded Publisher/Subscriber Architectures," the Journal of Systems and Software, Special Issue on Software Architecture – Engineering Quality Attributes, edited by Jan Bosch and Lars Lundberg, October 2002.

J37 Douglas C. Schmidt, "R&D Advances in Middleware for Distributed, Real-time and Embedded Systems," Communications of the ACM, Volume 45, Number 6, June 2002, edited by Gul Agha.

J36 Carlos O'Ryan, Douglas C. Schmidt, and J. Russell Noseworthy, "Patterns and Performance of a CORBA Event Service for Large-scale Distributed Interactive Simulations," International Journal of Computer Systems Science and Engineering, CRL Publishing, Volume 17, Number 2, March, 2002.

J35 Douglas C. Schmidt, Bala Natarajan, Aniruddha Gokhale, Nanbor Wang, and Chris Gill, "TAO: A Pattern-Oriented Object Request Broker for Distributed Real-time and Embedded Systems," IEEE Distributed Systems Online, Volume 3, Number 2, February, 2002.

J34 Douglas C. Schmidt, Rick Schantz, Mike Masters, Joseph Cross, David Sharp, and Lou DiPalma, "Towards Adaptive and Reflective Middleware for Network-Centric Combat Systems," CrossTalk, November, 2001.

J33 Nanbor Wang, Douglas C. Schmidt, Ossama Othman, and Kirthika Parameswaran, "Evaluating Meta-Programming Mechanisms for ORB Middleware," *IEEE Communications Magazine* special issue on "Evolving Communications Software: Techniques and Technologies," co-edited by Bill Opdyke and Algirdas Pakstas, Volume 39, Number 10, October, 2001.

J32 Nanbor Wang, Douglas C. Schmidt, Kirthika Parameswaran, and Michael Kircher, "Towards a Reflective Middleware Framework for QoS-enabled CORBA Component Model Applications," *IEEE Distributed Systems Online* special issue on Reflective Middleware, July, 2001.

J31 Chris Gill, David Levine, and Douglas C. Schmidt, "The Design and Performance of a Real-Time CORBA Scheduling Service," The *Real-time Systems, The International Journal of Time-Critical Computing Systems*, special issue on Real-Time Middleware, Kluwer Academic Publishers, guest editor Wei Zhao, Volume 20, Number 2, March 2001.

J30 Douglas C. Schmidt, Sumedh Mungee, Sergio Flores-Gaitan, and Aniruddha Gokhale, "Software Architectures for Reducing Priority Inversion and Non-determinism in Real-time Object Request Brokers," *Journal of Real-time Systems*, Kluwer, Vol. 21, No. 2, 2001.

J29 Ossama Othman, Carlos O'Ryan, and Douglas C. Schmidt, "An Efficient Adaptive Load Balancing Service for CORBA," *IEEE Distributed Systems Online*, March, 2001.

J28 Ossama Othman, Carlos O'Ryan, and Douglas C. Schmidt "The Design of an Adaptive CORBA Load Balancing Service, *IEEE Distributed Systems Online*, April, 2001.

J27 Carlos O'Ryan, Douglas C. Schmidt, Fred Kuhns, Marina Spivak, Jeff Parsons Irfan Pyarali, and David L. Levine, "Evaluating Policies and Mechanisms to Support Distributed Real-Time Applications with CORBA," *Concurrency and Computation: Practice and Experience* (Special Issue on Distributed Objects and Applications), Wiley and Sons, Vol. 13, No. 2, February, 2001.

J26 Douglas C. Schmidt, Vishal Kachroo, Yamuna Krisnamurthy, and Fred Kuhns, "Developing Next-generation Distributed Applications with QoS-enabled DPE Middleware," *IEEE Communications magazine*, edited by Abdi Modarressi and Sheshadri Mohan, Vol 17, No. 10, October, 2000.

J25 Douglas C. Schmidt and Fred Kuhns, "An Overview of the Real-time CORBA Specification," *IEEE Computer*, Special Issue on Object-Oriented Real-time Distributed Computing, edited by Eltefaat Shokri and Philip Sheu, June 2000.

J24 James Hu and Douglas C. Schmidt, Developing Flexible and High-performance Web Servers with Frameworks and Patterns, Symposium on Frameworks, *ACM Computing Surveys*, (Fayad and Wegner, eds.) Vol. 32(1es), March 2000.

J23 Fred Kuhns, Douglas C. Schmidt, Carlos O'Ryan, and David L. Levine, "Supporting High-performance I/O in QoS-enabled ORB Middleware,"

Cluster Computing: the Journal on Networks, Software, and Applications, Volume 3, Number 3, 2000.

J22 Irfan Pyarali, Carlos O'Ryan, Douglas C. Schmidt, Nanbor Wang, Vishal Kachroo, and Aniruddha Gokhale, "Using Principle Patterns to Optimize Real-time ORBs," *IEEE Concurrency*, Object-Oriented Systems track, edited by Murthy Devarakonda, Volumn 8, Number 1, January-March 2000.

J21 James Hu, Irfan Pyarali, and Douglas C. Schmidt, "The Object-Oriented Design and Performance of JAWS: A High-performance Web Server Optimized for High-speed Networks," The *Parallel and Distributed Computing Practices* journal, special issue on Distributed Object-Oriented Systems, edited by Maria Cobb, Vol. 3, No. 1, March 2000.

J20 Andy Gokhale and Douglas C. Schmidt, "Optimizing a CORBA IIOP Protocol Engine for Minimal Footprint Multimedia Systems," *IEEE Journal on Selected Areas in Communications* special issue on Service Enabling Platforms for Networked Multimedia Systems, September, 1999.

J19 Douglas C. Schmidt and Chris Cleeland, "Applying Patterns to Develop Extensible and Maintainable ORB Middleware," *IEEE Communications Magazine*, April, 1999.

J18 Irfan Pyarali and Douglas C. Schmidt, "An Overview of the CORBA Portable Object Adapter," Special Issue on CORBA in the ACM *StandardView* magazine, March, 1999.

J17 Prashant Jain, Seth Widoff, and Douglas C. Schmidt, "The Design and Performance of MedJava, A Distributed Electronic Medical Imaging System Developed with Java Applets and Web Tools," IEE/BCS *Distributed Systems Engineering Journal*, Volume 5, No. 4, December 1998.

J16 Douglas C. Schmidt, "Evaluating Architectures for Multi-threaded CORBA Object Request Brokers," *Communications of the ACM*, Special Issue on CORBA, ACM, edited by Krishnan Seetharaman, Volume 41, No. 10, October 1998.

J15 Andy Gokhale and Douglas C. Schmidt, "Measuring and Optimizing CORBA Latency and Scalability Over High-speed Networks," *IEEE Transactions on Computing*, April, 1998.

J14 Douglas C. Schmidt and James Hu, "Developing Flexible and High-performance Web Servers with Frameworks and Patterns," *Computing Surveys*, ACM, Vol. 29, March 1998.

J13 Douglas C. Schmidt, David Levine, and Sumedh Mungee, "The Design of the TAO Real-Time Object Request Broker," *Computer Communications*, Special Issue on Building Quality of Service into Distributed System, Elsevier Science, April, 1998.

J12 Guru Parulkar, Douglas C. Schmidt, Eileen Kraemer, Jon Turner, Anshul Kantawala, "An Architecture for Monitoring, Visualization, and Control and Gigabit Networks," *IEEE Network*, September/October, 1997.

J11 Douglas C. Schmidt, "Lessons Learned Building Reusable OO Frameworks for Distributed Software," *Communications of the ACM* Special Issue on OO Application Frameworks, ACM, Vol. 40, No. 10, October, 1997.

J10 Douglas C. Schmidt, "Applying Patterns to Meet the Challenges of Concurrent Software," *IEEE Concurrency*, Special Edition on Software Engineering for Parallel and Distributed Systems, Vol. 5, No. 3, August, 1997.

J9 Douglas C. Schmidt, Andy Gokhale, Tim Harrison, and Guru Parulkar, "A High-performance Endsystem Architecture for Real-time CORBA," *IEEE Communications Magazine*, Vol. 14, No. 2, February, 1997.

J8 Silvano Maffeis and Douglas C. Schmidt, "Constructing Reliable Distributed Communication Systems with CORBA," *IEEE Communications Magazine*, Vol. 14, No. 2, February, 1997.

J7 Douglas C. Schmidt, "Using Design Patterns to Develop Reuseable Object-Oriented Software," *ACM Computing Surveys*, Vol. 28, No. 4es, December 1996.

J6 Irfan Pyarali, Douglas C. Schmidt, and Tim Harrison, "Design and Performance of an Object-Oriented Framework for High-Speed Electronic Medical Imaging," *USENIX Computing Systems*, November/December, Vol. 9, No. 4, 1996.

J5 Douglas C. Schmidt, "A Family of Design Patterns for Application-level Gateways," *The Journal of Theory and Practice of Object Systems* (Special Issue on Patterns and Pattern Languages), Wiley and Sons, Vol. 2, No. 1, 1996.

J4 Douglas C. Schmidt, "Experience Using Design Patterns to Develop Reuseable Object-Oriented Communication Software," *Communications of the ACM Special Issue on Object-Oriented Experiences*, ACM, Vol. 38, No. 10, October, 1995, pp 65–74.

J3 Douglas C. Schmidt and Tatsuya Suda, "An Object-Oriented Framework for Dynamically Configuring Extensible Distributed Systems," *Distributed Systems Engineering Journal* (Special issue on Configurable Distributed Systems), IEE, Vol. 2, December, 1994, pp. 280–293.

J2 Douglas C. Schmidt, Donald F. Box, and Tatsuya Suda, "ADAPTIVE: A Dynamically Assembled Protocol Transformation, Integration, and eValuation Environment," *Journal of Concurrency: Practice and Experience*, Wiley and Sons, Ltd., Vol. 5, No. 4, June, 1993, pp. 269–286.

J1 Douglas C. Schmidt and Tatsuya Suda, "Transport System Architecture Services for High-Performance Communication Systems," *Journal of Selected Areas of Communications special-issue on Protocols for Gigabit Networks*, IEEE, Vol. 11, No. 4, May, 1993, pp. 489–506.

- **Book Publications and Book Chapters**

  - **Books Authored**

BA5 Frank Buschmann, Kevlin Henney, and Douglas C. Schmidt, *Pattern-Oriented Software Architecture: On Patterns and Pattern Languages*, Wiley and Sons, 2007.

BA4 Frank Buschmann, Kevlin Henney, and Douglas C. Schmidt, *Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing*, Wiley and Sons, 2007.

BA3 Steve Huston and Douglas C. Schmidt, *C++ Network Programming: Systematic Reuse with ACE and Frameworks*, Addison-Wesley Longman, 2003.

BA2 Douglas C. Schmidt and Steve Huston, *C++ Network Programming: Mastering Complexity with ACE and Patterns*, Addison-Wesley Longman, 2002.

BA1 Douglas C. Schmidt, Michael Stal, Hans Rohert, and Frank Buschmann, *Pattern-Oriented Software Architecture: Patterns for Concurrent and Networked Objects*, John Wiley and Sons, 2000.

- **Book-length Reports Authored**

BR4 Werner Dahm, Douglas Schmidt, et al. "Cyber Situational Awareness," SAB-TR-12-01, 1 October 2012.

BR3 Alan Eckbreth, Charles Saff, Kevin Connolly, Natalie Crawford, Chris Eick, Mark Goorsky, Neil Kacena, David Miller, Robert Schafrik, Douglas Schmidt, "Sustaining Air Force Aging Aircraft into the 21st Century," SAB-TR-11-01 1 August 2011.

BR2 William Scherlis, Robert Behler, Barry Boehm, Lori Clarke, Michael Cusumano, Mary Ann Davidson, Larry Druffel, Russell Frew, James Larus, Greg Morrisett, Walker Royce, Douglas C. Schmidt, John Stenbit, Kevin Sullivan, *Critical Code Software Producibility for Defense*, Committee for Advancing Software-Intensive Systems Producibility, National Research Council of the National Academies, Washington, D.C.

BR1 Linda Northrop, Peter Feiler, Richard P. Gabriel, John Goodenough, Rick Linger, Tom Longstaff, Rick Kazman, Mark Klein, Linda Northrop, Douglas Schmidt, Kevin Sullivan, and Kurt Wallnau *Ultra-Large-Scale Systems: Software Challenge of the Future*, Software Engineering Institute, June 2006.

- **Books Edited**

BE3 Co-editor of *Building Application Frameworks: Object-Oriented Foundations of Framework Design*, John Wiley & Sons, 1999 (co-editors are Mohamed Fayad and Ralph Johnson), ISBN 0-471-24875-4

BE2 Co-editor of *Implementing Application Frameworks: Object-Oriented Frameworks at Work*, John Wiley & Sons, 1999 (co-editors are Mohamed Fayad and Ralph Johnson), ISBN 0-471-25201-8

BE1 Co-editor of *Pattern Languages of Program Design*, Addison-Wesley, 1995 (co-editor is Jim Coplien, Bell Labs).

- **Book Chapters**

BC64 Joseph K. Cross and Douglas C. Schmidt, "Quality Connector: A Pattern Language for Provisioning and Managing Quality-Constrained Services in Distributed Real-time and Embedded Systems," *Design Patterns for Distributed and Real-time Systems*, edited by Christopher Gill and Lisa DiPippo, Kluwer Academic Publisher (to appear).

BC63 Angelo Corsaro, Douglas C. Schmidt, Raymond Klefstad, and Carlos O'Ryan, "Virtual Component: a Design Pattern for Memory-Constrained Embedded Applications," *Design Patterns for Distributed and Real-time Systems*, edited by Christopher Gill and Lisa DiPippo, Kluwer Academic Publishers (to appear).

BC62 Michael Stal and Douglas C. Schmidt, "Activator," *Design Patterns for Distributed and Real-time Systems*, edited by Christopher Gill and Lisa DiPippo, Kluwer Academic Publishers (to appear).

BC61 Michael Walker, Douglas C. Schmidt, and Jules White, "An Elastic Platform for Large-scale Assessment of Software Assignments for MOOCs (EPLASAM)," ... 2015 (to appear).

BC60 Joseph Hoffert, Douglas C. Schmidt, and Aniruddha Gokhale, "Quantitative Productivity Analysis of a Domain-Specific Modeling Language," in the *Handbook of Research on Innovations in Systems and Software Engineering*, IGI Global, Aug 2014, pp. 313-344.

BC59 Will Otte, Douglas C. Schmidt, and Aniruddha Gokhale, "Performance and Scalability of a Large-scale Deployment and Configuration Framework," *The Performance of Open Source Applications*, edited by Tony Arkles and Tavish Armstrong, O'Reilly, 2013.

BC58 Chris Thompson, Jules White, and Douglas C. Schmidt, "Analyzing Mobile Application Software Power Consumption via Model-Driven Engineering," *Advances and Applications in Model-Driven Software Engineering*, edited by Vicente Garcia Diaz, IGI Global, 2013.

BC57 James Edmondson and Douglas C. Schmidt, "Towards Accurate Simulation of Large-Scale Systems via Time Dilation," *Real-time Simulation Technologies: Principles, Methodologies, and Applications*, edited by Katalin Popovici and Pieter J. Mosterman, CRC Press, 2012.

BC56 James Hill and Douglas C. Schmidt, "Using Test Clouds to Enable Continuous Integration Testing of Distributed Real-time and Embedded System Applications," *Software Testing in the Cloud: Perspectives on an Emerging Discipline*. Edited by Scott Tilley and Tauhida Parveen, IGI Global, 2012.

BC55 Angelo Corsaro and Douglas C. Schmidt, "The Data Distribution Service: The Communication Middleware Fabric for Scalable and

Extensible Systems-of-Systems," *System of Systems*, edited by Adrian V. Gheorghe, InTech, 2012.

BC54 Joe Hoffert and Douglas C. Schmidt, "Improving Software Development Productivity for QoS Policy Configurations," *Model-Driven Domain Analysis and Software Development: Architectures and Functions*, a book edited by Janis Osis and Erika Asnina, 2011.

BC53 Nilabja Roy and Douglas C. Schmidt, "Model-Driven Performance Evaluation of Web Application Portals," *Model-Driven Domain Analysis and Software Development: Architectures and Functions*, a book edited by Janis Osis and Erika Asnina, 2011.

BC52 Brian Dougherty, Jules White, and Douglas C. Schmidt, "MDA-based Configuration of Distributed Real-time and Embedded Systems," *Model-Driven Domain Analysis and Software Development: Architectures and Functions*, a book edited by Janis Osis and Erika Asnina, 2011.

BC51 Hamilton Turner, Jules White, Brian Dougherty, and Douglas C. Schmidt, "Building Mobile Sensor Networks Using Smartphones and Web Services: Ramifications and Development Challenges," *Handbook of Research on Mobility and Computing: Evolving Technologies and Ubiquitous Impacts*, edited by Maria Manuela Cruz-Cunha and Fernando Moreira, IGI Global, Hershey, PA, USA 2009.

BC50 Gan Deng, Jeff Gray, Douglas C. Schmidt, Yuehua Lin, Aniruddha Gokhale, and Gunther Lenz, "Evolution in Model-Driven Software Product-Line Architectures," *Software Applications: Concepts, Methodologies, Tools, and Applications*, edited by Pierre F. Tiako, 2009.

BC49 Jules White, Douglas C. Schmidt, Andrey Nechypurenko, and Egon Wuchner, "Reducing the Complexity of Modeling Large Software Systems," *Software Applications: Concepts, Methodologies, Tools, and Applications*, edited by Pierre F. Tiako, 2009.

BC48 Jules White, Brian Doughtery, Harrison Strowd, and Douglas C. Schmidt, "Using Filtered Cartesian Flattening and Microrebooting to Build Enterprise Applications with Self-adaptive Healing," *Software Engineering for Self-Adaptive Systems*, edited by Betty H. C. Cheng, Rogerio de Lemos, Holger Giese, Paola Inverardi, and Jeff Magee, 2009.

BC47 Jeff Gray, Sandeep Neema, Jing Zhang, Yuehua Lin, Ted Bapty, Aniruddha Gokhale, and Douglas C. Schmidt, "Concern Separation for Adaptive QoS Modeling in Distributed Real-Time Embedded Systems," *Behavioral Modeling for Embedded Systems and Technologies: Applications for Design and Implementation*, edited by Joa M. Fernandes and Luis Gomes, 2009.

BC46 Rick Schantz and Douglas C. Schmidt, "Middleware for Distributed Systems," Wiley *Encyclopedia of Computer Science and Engineering*, edited by Benjamin Wah, 2008.

B45 Jules White, Andrey Nechypurenko, Egon Wuchner, and Douglas C. Schmidt, "Reducing the Complexity of Optimizing Large-scale Systems by Integrating Constraint Solvers with Graphical Modeling Tools," *Designing Software-Intensive Systems: Methods and Principles*, Edited by Dr. Pierre F. Tiako, Langston University, OK, April 2008.

B44 Gan Deng, Douglas C. Schmidt, Aniruddha Gokhale, Jeff Gray, Yuehua Lin, and Gunther Lenz, "Evolution in Model-Driven Software Product-line Architecture, *Designing Software-Intensive Systems: Methods and Principles*, Edited by Dr. Pierre F. Tiako, Langston University, OK, April 2008.

B43 Daniel G. Waddington, Nilabja Roy, and Douglas C. Schmidt, "Dynamic Analysis and Profiling of Multi-threaded Systems," *Designing Software-Intensive Systems: Methods and Principles*, Edited by Dr. Pierre F. Tiako, Langston University, OK, April 2008.

B42 Krishnakumar Balasubramanian, Douglas C. Schmidt, Zoltan Molnar, and Akos Ledeczi, "System Integration via Model-Driven Engineering," *Designing Software-Intensive Systems: Methods and Principles* Edited by Dr. Pierre F. Tiako, Langston University, OK, April 2008.

B41 James Hill, Douglas C. Schmidt,, and John Slaby, "System Execution Modeling Tools for Evaluating the Quality of Service of Enterprise Distributed Real-time and Embedded System",*Designing Software-Intensive Systems: Methods and Principles*, Edited by Dr. Pierre F. Tiako, Langston University, OK, April 2008.

B40 Gan Deng, Chris Gill, Douglas C. Schmidt, and Nanbor Wang, "QoS-enabled Component Middleware for Distributed Real-Time and Embedded Systems," Handbook of Real-Time and Embedded Systems (I. Lee, J. Leung, and S. Son, eds.), CRC Press, 2007.

B39 William Otte and Douglas C. Schmidt, "Labor-Saving Architecture: an Object-Oriented Framework for Networked Software," *Beautiful Code*, edited by Greg Wilson, O'Reilly, 2007.

BC38 Irfan Pyarali, Carlos O'Ryan, and Douglas C. Schmidt, "A Pattern Language for Efficient, Predictable, Scalable, and Flexible Dispatching Components," Pattern Language of Program Design 5 book, Addison-Wesley, Reading, MA, 2006.

BC37 Douglas C. Schmidt, Krishnakumar Balasubramanian, Arvind S. Krishna, Emre Turkay, and Aniruddha Gokhale, Model-driven Development of Component-based Distributed Real-time and Embedded Systems, Model Driven Engineering for Distributed Real-

time and Embedded Systems, edited by Sebastien Gerard, Joel Champea, and Jean-Philippe Babau, Hermes, 2005.

BC36 Gabriele A. Trombetti, Aniruddha Gokhale, Douglas C. Schmidt, John Hatcliff, Gurdip Singh, and Jesse Greenwald, "An Integrated Model-driven Development Environment for Composing and Validating Distributed Real-time and Embedded Systems," *Model-driven Software Development: Volume II of Research and Practice in Software Engineering*, edited by Sami Beydeda, Matthias Book, and Volker Gruhn, Springer-Verlag, 2005.

BC35 Arvind S. Krishna, Douglas C. Schmidt, Ray Klefstad, and Angelo Corsaro, "Real-time CORBA Middleware," in *Middleware for Communications*, edited by Qusay Mahmoud, Wiley and Sons, New York, 2003.

BC34 Nanbor Wang, Douglas C. Schmidt, Aniruddha Gokhale, Craig Rodrigues, Balachandran Natarajan, Joseph P. Loyall, Richard E. Schantz, and Christopher D. Gill, "QoS-enabled Middleware," in *Middleware for Communications*, edited by Qusay Mahmoud, Wiley and Sons, New York, 2003.

BC33 Aniruddha Gokhale, Douglas C. Schmidt, Balachandran Natarajan, Jeff Gray, and Nanbor Wang, "Model Driven Middleware," in *Middleware for Communications*, edited by Qusay Mahmoud, Wiley and Sons, New York, 2003.

BC32 Jeff Gray, Janos Sztipanovits, Ted Bapty, Sandeep Neema, Aniruddha Gokhale, and Douglas C. Schmidt, "Two-level Aspect Weaving to Support Evolution of Model-Based Software," in *Aspect-Oriented Software Development,* edited by Robert Filman, Tzilla Elrad, Mehmet Aksit, and Siobhan Clarke, Reading, Massachusetts: Addison-Wesley, 2003.

BC31 Joseph K. Cross and Douglas C. Schmidt "Applying the Quality Connector Pattern to Optimize Distributed Real-time and Embedded Middleware," *Patterns and Skeletons for Parallel and Distributed Computing*, edited by Fethi Rabhi and Sergei Gorlatch, Springer Verlag, 2002.

BC30 Richard E. Schantz and Douglas C. Schmidt, "Middleware for Distributed Systems: Evolving the Common Structure for Network-centric Applications," *Encyclopedia of Software Engineering*, edited by John Marciniak and George Telecki, Wiley and Sons, 2001.

BC29 Sumedh Mungee, Nagarajan Surendran, Yamuna Krishnamurthy, and Douglas C. Schmidt "The Design and Performance of a CORBA Audio/Video Streaming Service," *Multimedia Networking: Technology, Management, and Applications*, edited by Mahbubur Syed, Idea Group Publishing, Hershey, USA, 2001.

BC28 Nanbor Wang, Douglas C. Schmidt, and Carlos O'Ryan "An Overview of the CORBA Component Model," *Component-Based Software*

Engineering, (George Heineman and Bill Councill, eds.), Addison-Wesley, Reading, MA, 2001.

BC27 Douglas C. Schmidt, "Applying a Pattern Language to Develop Application-Level Gateways," Design Patterns in Communications, (Linda Rising, ed.), Cambridge University Press, 2000.

BC26 Douglas C. Schmidt and Chris Cleeland, "Applying a Pattern Language to Develop Extensible ORB Middleware," Design Patterns in Communications, (Linda Rising, ed.), Cambridge University Press, 2000.

BC25 Carlos O'Ryan, Fred Kuhns, Douglas C. Schmidt, and Jeff Parsons, "Applying Patterns to Develop a Pluggable Protocols Framework for Object Request Broker Middleware," Design Patterns in Communications, (Linda Rising, ed.), Cambridge University Press, 2000.

BC24 David L. Levine, Christopher D. Gill, and Douglas C. Schmidt, "Object Lifecycle Manager – A Complementary Pattern for Controlling Object Creation and Destruction," Design Patterns in Communications, (Linda Rising, ed.), Cambridge University Press, 2000.

BC23 Douglas C. Schmidt, "A Family of Design Patterns For Flexibly Configuring Network Services in Distributed Systems," Design Patterns in Communications, (Linda Rising, ed.), Cambridge University Press, 2000.

BC22 James Hu and Douglas C. Schmidt, "JAWS: A Framework for High-performance Web Servers," *Object-Oriented Application Frameworks* book, John Wiley & Sons, October, 1999.

BC21 Chris Cleeland and Douglas C. Schmidt, "External Polymorphism, An Object Structural Pattern for Transparently Extending C++ Concrete Data Types" in C++ Gems II, (Robert Martin, ed.), SIGS, NY, 1999.

BC20 Douglas C. Schmidt, "GPERF: A Perfect Hash Function Generator" in C++ Gems II, (Robert Martin, ed.), SIGS, NY, 1999.

BC19 Douglas C. Schmidt, Tim H. Harrison, and Nat Pryce, "Thread-specific Storage: an Object Behavioral Pattern for Efficiently Accessing per-Thread State" in C++ Gems II, (Robert Martin, ed.), SIGS, NY, 1999.

BC18 Irfan Pyarali, Tim Harrison, Douglas C. Schmidt, and Thomas Jordan, "Proactor: an Object Behavioral Pattern for Demultiplexing and Dispatching Handlers for Asynchronous Events," *Pattern Languages of Program Design,* (Harrison, Foote, and Rohnert, eds.), Addison-Wesley, Reading, MA, 1999.

BC17 Douglas C. Schmidt and Paul Stephenson, "Using Design Patterns to Evolve System Software from UNIX to Windows NT," In *The Patterns Handbook*, (Linda Rising, ed.), Cambridge University Press, 1998.

BC16 Douglas C. Schmidt, David L. Levine, and Chris Cleeland, "Architectures and Patterns for High-performance, Real-time CORBA

Object Request Brokers," *Advances in Computers*, Academic Press, Ed., Marvin Zelkowitz, Volume 48, July 1999.

BC15 Douglas C. Schmidt and Tatsuya Suda, "A Framework for Measuring the Performance of Alternative Process Architectures for Parallel Communication Subsystems," in *Network Performance Modeling and Simulation*, Walrand, Bagchi, and Zobrist, Ed., Gordon and Breach Publishers, 1998.

BC14 Douglas C. Schmidt, "Applying Design Patterns and Frameworks to Develop Object-Oriented Communication Software," The Handbook of Programming Languages, Volume I, edited by Peter Salus, MacMillan Computer Publishing, 1997.

BC13 Chris Cleeland, Douglas C. Schmidt, and Tim H. Harrison, "External Polymorphism – An Object Structural Pattern for Transparently Extending Concrete Data Types," *Pattern Languages of Program Design,* (Martin, Buschmann, and Riehl, eds.), Addison-Wesley, Reading, MA, 1997.

BC12 Timothy H. Harrison, Douglas C. Schmidt, and Irfan Pyarali, "Asynchronous Completion Token – An Object Behavioral Pattern for Efficient Asynchronous Event Handling," *Pattern Languages of Program Design,* (Martin, Buschmann, and Riehl, eds.), Addison-Wesley, Reading, MA, 1997.

BC11 Douglas C. Schmidt and Timothy H. Harrison, "Double-Checked Locking – A Optimization Pattern for Efficiently Initializing and Accessing Thread-safe Objects," *Pattern Languages of Program Design,* (Martin, Buschmann, and Riehl, eds.), Addison-Wesley, Reading, MA, 1997.

BC10 Douglas C. Schmidt, "Acceptor and Connector – A Family of Object Creational Patterns for Initializing Communication Services," *Pattern Languages of Program Design,* (Martin, Buschmann, and Riehl, eds.), Addison-Wesley, Reading, MA, 1997.

BC9 Douglas C. Schmidt and Paul Stephenson, "Using Design Patterns to Evolve System Software from UNIX to Windows NT," In *Wisdom of the Gurus*, (Charles Bowman, ed.), Cambridge University Press, 1996.

BC8 Douglas C. Schmidt and Steve Vinoski, "Comparing Alternative Distributed Programming Techniques" in *Wisdom of the Gurus*, (Charles Bowman, ed.), Cambridge University Press, 1996.

BC7 Douglas C. Schmidt, "A Case Study in C++ Design Evolution" in *C++ Gems,* (Stanley Lippman, ed.), SIGS, NY, 1996, pp. 99–120.

BC6 Douglas C. Schmidt and Steve Vinoski, "Distributed Object Computing in C++" in *C++ Gems,* (Stanley Lippman, ed.), SIGS, NY, 1996, pp. 303–316.

BC5 Douglas C. Schmidt and Steve Vinoski, "Comparing Alternative Distributed Programming Techniques" in *C++ Gems,* (Stanley Lippman, ed.), SIGS, NY, 1996, pp. 316–336.

BC4 Douglas C. Schmidt and Steve Vinoski, "Comparing Alternative Server Programming Techniques" in *C++ Gems,* (Stanley Lippman, ed.), SIGS, NY, 1996, pp. 337–362.

BC3 Douglas C. Schmidt and Charles D. Cranor, "Half-Sync/Half-Async: A Architectural Pattern for Efficient and Well-structured Concurrent I/O" in *Pattern Languages of Program Design,* (Coplien, Vlissides, and Kerth, eds.), Addison-Wesley, Reading, MA, 1996.

BC2 R. Greg Lavender and Douglas C. Schmidt, "Active Object: An Object Behavioral Pattern for Concurrent Programming," in *Pattern Languages of Program Design,* (Coplien, Vlissides, and Kerth, eds.), Addison-Wesley, Reading, MA, 1996.

BC1 Douglas C. Schmidt, "Reactor: An Object Behavioral Pattern for Concurrent Event Demultiplexing and Event Handler Dispatching," *Pattern Languages of Program Design,* (Addison-Wesley, 1995), edited by James O. Coplien and Douglas C. Schmidt.

- **Refereed Conference Publications**

C180 Nickolas H. Guertin, Robert Sweeney, and Douglas C. Schmidt, "How the Navy Can Use Open Systems Architecture to Revolutionize Capability Acquisition," Naval Postgraduate School's Acquisition Research Symposium, May 13th 2015, Monterey, CA.

C179 Nickolas H. Guertin, Robert Sweeney, and Douglas C. Schmidt, "Benefits of Applying an Open Systems Architecture Approach," 17th Systems Engineering Conference, National Defense Industry Association, October 27-30, 2014, Washington DC.

C178 Kyoungho An, Aniruddha Gokhale, Sumant Tambe, Gerardo Pardo-Castellote, and Douglas C. Schmidt, "Content-based Filtering Discovery Protocol (CFDP): Scalable and Efficient OMG DDS Discovery Protocol," 8th ACM International Conference on Distributed Event-Based Systems, Mumbai, India, May 26-29, 2014.

C177 Balakrishnan Dasarathy, Kevin Sullivan, Douglas C. Schmidt, Douglas H. Fisher, and Adam Porter, "The Past, Present, and Future of MOOCs and Their Relevance to Software Engineering," 36th ACM/IEEE International Conference on Software Engineering, Hyderabad, India, May 31 - June 7th, 2014.

C176 Rick Leathart, Adam Porter, Douglas Schmidt, Michael O'Hare, Harry Crisp, and Barry Laird, "Capability-Based Technical Reference Frameworks for Open System Architecture Implementations," Systems Engineering Conference (SEDC) 20134, Washington DC, April 3rd and 4th, 2014, Chantilly, VA.

C175 Douglas C. Schmidt, Chris Gill, and Jules White, "Elastic Infrastructure to Support Computing Clouds for Large-scale Cyber-Physical Systems," Proceedings of the International Symposium on Object-Oriented Real-time Distributed Computing (ISORC), June 2014, Reno, Nevada.

C174 Douglas C. Schmidt and Zach McCormick, "Creating and Teaching a MOOC on Pattern-Oriented Software Architecture for Concurrent and Networked Software," Proceedings of the WaveFront Forum at the SPLASH 2013 conference, October 2013, Indianapolis, IN.

C173 Zach McCormick and Douglas C. Schmidt, "Data Synchronization Patterns in Mobile Application Design," Proceedings of the Pattern Languages of Programs (PLoP) 2012 conference, October 19-21, Tucson, Arizona.

C172 James Edmondson, Aniruddha Gokhale and Douglas Schmidt, "Approximation Techniques for Maintaining Real-time Deployments Informed by User-provided Dataflows Within a Cloud," 31st International Symposium on Reliable Distributed Systems (SRDS 2012), 8th-11th October 2012, Irvine, California.

C171 James Edmondson, Douglas C. Schmidt, and Aniruddha Gokhale "QoS-enabled Distributed Mutual Exclusion in Public Clouds," Proceedings of the 1st International Symposium on Secure Virtual Infrastructures (DOA-SVI'11), October 17-19, 2011, Crete, Greece.

C170 Brian Dougherty, Jules White, Russell Kegley, Jonathan Preston, Douglas C. Schmidt, and Aniruddha Gokhale, "Optimizing Integrated Application Performance with Cache-aware Metascheduling," Proceedings of the 1st International Symposium on Secure Virtual Infrastructures (DOA-SVI'11), October 17-19, 2011, Crete, Greece.

C169 Akram Hakiri, Aniruddha Gokhale, Douglas C. Schmidt, Berthou Pascal, Joe Hoffert, and Gayraud Thierry, "A SIP-based Network QoS Provisioning Framework for Cloud-hosted DDS Applications," Proceedings of the 1st International Symposium on Secure Virtual Infrastructures (DOA-SVI'11), October 17-19, 2011, Crete, Greece.

C168 James Hill and Douglas C. Schmidt, "Experiences with Service-Oriented Middleware for Dynamic Instrumentation of Enterprise Distributed Real-time and Embedded Systems," Proceedings of the 1st International Symposium on Secure Virtual Infrastructures (DOA-SVI'11), October 17-19, 2011, Crete, Greece.

C167 Will Otte, Aniruddha Gokhale, Douglas C. Schmidt, and Johnny Willemsen, "Infrastructure for Component-based DDS Application Development," proceedings of the Tenth International Conference on Generative Programming and Component Engineering (GPCE'11), October 22-23, 2011 Portland, Oregon, USA.

C166 Will Otte, Aniruddha Gokhale, and Douglas C. Schmidt, Techniques for Predictable Deployment Latencies in Large-scale Component-based Distributed Real-time and Embedded Systems, Proceedings of the 14th International ACM SIGSOFT Symposium on Component Based Software Engineering (CBSE-2011), June 21th - 23th, 2011, in Boulder, Colorado, USA.

C165 Chris Thompson, Hamilton Turner, Jules White, and Douglas C. Schmidt, Analyzing Mobile Application Software Power Consumption via Model-

Driven Engineering, Proceedings of the 1st International Conference on Pervasive and Embedded Computing and Communication Systems, Algarve, Portugal, March 5-7, 2011.

C164 Joe Hoffert, Douglas C. Schmidt, Aniruddha Gokhale, "Adapting Distributed Real-time and Embedded Publish/Subscribe Middleware for Cloud-Computing Environments," Proceedings of the ACM/IFIP/USENIX 11th International Middleware Conference, Bangalore, India, November 30-Dec 3, 2010.

C163 Joe Hoffert and Douglas Schmidt, "Evaluating Supervised Machine Learning for Adapting Enterprise DRE Systems," Proceedings of the International Symposium on Intelligence Information Processing and Trusted Computing (IPTC 2010), Huanggang City, China, October 28-29, 2010.

C162 J. Benjamin Gotow, Krzysztof Zienkiewicz, Jules White, and Douglas C. Schmidt, "Addressing Challenges in Delivering Augmented Reality Applications to Smartphones," Proceedings of the Third International ICST Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications (Mobilware 2010), June 30-July 2, 2010, Chicago, IL.

C161 Chris Thompson, Jules White, Brian Dougherty, Adam Albright, and Douglas C. Schmidt, "Using Smartphones and Wireless Mobile Networks to Detect Car Accidents and Provide Situational Awareness to Emergency Responders," Proceedings of the Third International ICST Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications (Mobilware 2010), June 30-July 2, 2010, Chicago, IL.

C160 James H. Hill, Hunt Sutherland, Douglas C. Schmidt, Thomas Silveria, John M. Slaby, Paul Staudinger, and Nikita A. Visnevski, "OASIS: A Service-Oriented Architecture for Dynamic Instrumentation of Enterprise Distributed Real-time and Embedded Systems," Proceedings of the 13th International Symposium on Object/Component/Service-oriented Real-time Distributed Computing (ISORC '10), May 5-6, 2010, Carmona, Spain.

C159 Joseph P. Loyall, Matthew Gillen, Aaron Paulos, Larry Bunch, Marco Carvalho, James Edmondson, Pooja Varshneya, Douglas C. Schmidt, Andrew Martignoni, "Dynamic Policy-Driven Quality of Service in Service-Oriented Systems," Proceedings of the 13th International Symposium on Object/Component/Service-oriented Real-time Distributed Computing (ISORC '10), May 5-6, 2010, Carmona, Spain.

C158 Jaiganesh Balasubramanian, Aniruddha Gokhale, Abhishek Dubey, Friedhelm Wolf, Chenyang Lu, Chris Gill, and Douglas C. Schmidt, "Middleware for Resource-Aware Deployment and Configuration of Fault-tolerant Real-time Systems," Proceedings of the 16th IEEE Real-Time and Embedded Technology and Applications Symposium Stockholm, Sweden, April 12 - 15, 2010.

C157 John S. Kinnebrew, Daniel L. C. Mack, Gautam Biswas, Douglas C. Schmidt, "Coordination of Planning and Scheduling Techniques for a Distributed, Multi-level, Multi-agent System", Proceedings of the 2nd International

Conference on Agents and Artificial Intelligence (ICAART), Valencia, Spain, January 22-24, 2010.

C156 Nilabja Roy, Yuan Xue, Aniruddha Gokhale, Larry Dowdy and Douglas C. Schmidt, "A Component Assignment Framework for Improved Capacity and Assured Performance inWeb Portals," Proceedings of the 11th International Symposium on Distributed Objects, Middleware, and Applications (DOA'09) Vilamoura, Algarve-Portugal, Nov 01 - 03, 2009.

C155 Joe Hoffert, Douglas C. Schmidt, and Aniruddha Gokhale, "Evaluating Transport Protocols for Real-time Event Stream Processing Middleware and Applications," Proceedings of the 11th International Symposium on Distributed Objects, Middleware, and Applications (DOA'09) Vilamoura, Algarve-Portugal, Nov 01 - 03, 2009.

C154 Joe Hoffert and Douglas C. Schmidt, "Maintaining QoS for Publish/Subscribe Middleware in Dynamic Environments," Fast Abstract, 3rd ACM International Conference on Distributed Event-Based Systems (DEBS 2009) July 6-9, 2009 - Nashville, TN, USA.

C153 Nilabja Roy, Larry Dowdy, and Douglas C. Schmidt, "The Impact of Variability on Soft Real-Time System Scheduling," Proceedings of the 15th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 2009), Beijing, China, August 24-26, 2009.

C152 Friedhelm Wolf, Jaiganesh Balasubramanian, Aniruddha Gokhale, and Douglas C. Schmidt, "Component Replication based on Failover Units," Proceedings of the 15th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 2009), Beijing, China, August 24-26, 2009.

C151 Jules White, David Benavides, Brian Dougherty, Douglas C. Schmidt, "Automated Reasoning for Multi-step Software Product-line Configuration Problems," Proceedings of the 13th International Software Product Line Conference (SPLC 2009) August 24-28, 2009, San Francisco, CA.

C150 Joseph Loyall, Marco Carvalho, Douglas Schmidt, Matthew Gillen, Andrew Martignoni III, Larry Bunch, James Edmondson, and David Corman, "QoS Enabled Dissemination of Managed Information Objects in a Publish-Subscribe-Query Information Broker," the SPIE Defense Transformation and Net-Centric Systems conference, April, 2009, Orlando, FL.

C149 Brian Dougherty, Jules White, Jaiganesh Balasubramanian, Chris Thompson, and Douglas C. Schmidt, "Deployment Automation with BLITZ," Proceedings of the Emerging Results track at the 31st International Conference on Software Engineering, Vancouver, Canada, May 16-24, 2009.

C148 Brian Dougherty, Jules White, Chris Thompson, and Douglas C. Schmidt, "Automating Hardware and Software Evolution Analysis," Proceedings of the 16th Annual IEEE International Conference and Workshop on the

Engineering of Computer Based Systems (ECBS), April 13-16, 2009, San Francisco, CA USA.

C147 James H. Hill, Hamilton A. Turner, James R. Edmondson, and Douglas C. Schmidt, "Unit Testing Non-functional Concerns of Component-based Distributed Systems," Proceedings of the 2nd International Conference on Software Testing, Verification, and Validation, April 1 - 4, 2009, Denver, Colorado.

C146 John S. Kinnebrew, William R. Otte, Nishanth Shankaran, Gautam Biswas, and Douglas C. Schmidt, "Intelligent Resource Management and Dynamic Adaptation in a Distributed Real-time and Embedded Sensor Web System," Proceedings of the 12th International Symposium on Object/Component/Service-oriented Real-time Distributed Computing (ISORC '09), Tokyo, Japan, March 17-20, 2009.

C145 Jaiganesh Balasubramanian, Sumant Tambe, Chenyang Lu, Aniruddha Gokhale, Christopher Gill, and Douglas C. Schmidt, "Adaptive Failover for Real-time Middleware with Passive Replication, Proceedings of the 15th Real-time and Embedded Applications Symposium (RTAS) 2009, San Francisco, CA, United States, April 13 - 16, 2009.

C144 William R. Otte, John. S. Kinnebrew, Douglas C. Schmidt, and Gautam Biswas, "A Flexible Infrastructure for Distributed Deployment in Adaptive Sensor Webs," Proceedings of the 2009 IEEE Aerospace Conference, Big Sky, Montana, March 2009.

C143 Nanbor Wang, Douglas C. Schmidt, Angelo Corsaro, and Hans Van'T Hag, "Toward an Adaptive Data Distribution Service for Dynamic Large-Scale Network-Centric Operation and Warfare Systems," Proceedings of the 2008 Military Communications Conference, November 17-19, 2008 in San Diego, CA.

C142 Joe Hoffert, Douglas C. Schmidt, and Aniruddha Gokhale, "DQML: A Modeling Language for Configuring Publish/Subscribe Quality of Service Policies," Proceedings of the Distributed Objects, Middleware, and Applications (DOA'08), Monterrey, Mexico, Nov 10 - 12, 2008.

C141 Nilabja Roy, Akshay Dabholkar, Nathan Hamm, Larry Dowdy and Douglas Schmidt, "Modeling Software Contention using Colored Petri Nets," Proceedings of the 16th Annual Meeting of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), September 8-10, Baltimore, MD.

C140 Jules White, Douglas C. Schmidt, David Benavides, Pablo Trinidad, Antonio Ruiz-Cortez, "Automated Diagnosis of Product-line Configuration Errors in Feature Models," Proceedings of the Software Product Lines Conference (SPLC), September, 2008, Limerick, Ireland.

C139 Jules White and Douglas C. Schmidt, "Model-Driven Product-Line Architectures for Mobile Devices," Proceedings of the 17th Annual Conference of the International Federation of Automatic Control, Seoul, Korea, July 6-11, 2008.

C138 Jules White and Douglas C. Schmidt, "Automated Configuration of Component-based Distributed Real-time and Embedded Systems from Feature Models," Proceedings of the 17th Annual Conference of the International Federation of Automatic Control, Seoul, Korea, July 6-11, 2008.

C137 William R. Otte, John. S. Kinnebrew, Douglas C. Schmidt, Gautam Biswas, and Dipa Suri, "Application of Middleware and Agent Technologies to a Representative Sensor Network," Proceedings of the 2008 Earth Science Technology Conference, June 24-26, 2008, University of Maryland.

C136 Jai Balasubramanian, Aniruddha Gokhale, Douglas C. Schmidt, and Nanbor Wang, "Towards Middleware for Fault-tolerance in Distributed Real-time and Embedded Systems," Proceedings of the 8th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS 2008), Oslo, Norway, June 4-6, 2008.

C135 Serena Fritsch, Aline Senart, Douglas C. Schmidt, and Siobhan Clarke, "Time-bounded Adaptation for Automotive System Software," Proceedings of the Experience Track on Automotive Systems at the 30th International Conference on Software Engineering, Leipzig, Germany, 10 - 18 May 2008.

C134 Nilabja Roy, John S. Kinnebrew, Nishanth Shankaran, Gautam Biswas, and Douglas C. Schmidt, "Toward Effective Multi-capacity Resource Allocation in Distributed Real-time and Embedded Systems," Proceedings of the 11th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing, Orlando, Florida, May 5-7, 2008.

C133 Gan Deng, Douglas C. Schmidt, and Aniruddha Gokhale, "CaDANCE: A Criticality-Aware Deployment And Configuration Engine," Proceedings of the 11th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing, Orlando, Florida, May 5-7, 2008.

C132 Krishnakumar Balasubramanian and Douglas C. Schmidt, "Physical Assembly Mapper: A Model-driven Optimization Tool for QoS-enabled Component Middleware," Proceedings of 14th IEEE Real-Time and Embedded Technology and Applications Symposium, St. Louis, MO, United States, April 22 - April 24, 2008.

C131 Jaiganesh Balasubramanian, Sumant Tambe, Balakrishnan Dasarathy, Shrirang Gadgil, Frederick Porter, Aniruddha Gokhale, and Douglas C. Schmidt, "NetQoPE: A Model-driven Network QoS Provisioning Engine for Distributed Real-time and Embedded Systems," Proceedings of 14th IEEE Real-Time and Embedded Technology and Applications Symposium, St. Louis, MO, United States, April 22 - April 24, 2008.

C130 James Hill, Douglas C. Schmidt, John Slaby, and Adam Porter, "CiCUTS: Combining System Execution Modeling Tools with Continuous Integration Environments," Proceedings of the 15th Annual IEEE International Conference and Workshops on the Engineering of Computer Based Systems (ECBS), March 31st - 4th April, 2008 Belfast, Northern Ireland.

C129 Vinny Cahill, Aline Senart, Douglas C. Schmidt, Stefan Weber, Anthony Harrington, Barbara Hughes, and Kulpreet Singh, "The Managed Motorway: Real-time Vehicle Scheduling: A Research Agenda," Proceedings of the ACM HotMobile 2008, Silverado Resort, Napa Valley, CA, USA February 25-26, 2008.

C128 Jules White, Krzysztof Czarnecki, Douglas Schmidt, Gunther Lenz, Christoph Wienands, Egon Wuchner, Ludger Fiege, "Automated Model-based Configuration of Enterprise Java Applications," The 11th IEEE International EDOC Conference (EDOC 2007), 15-19 October 2007, Annapolis, Maryland U.S.A.

C127 Joe Hoffert, Douglas Schmidt, and Aniruddha Gokhale, "A QoS Policy Configuration Modeling Language for Publish/Subscribe Middleware Platforms," Proceedings of International Conference on Distributed Event-Based Systems (DEBS), June 20-22nd, 2007, Toronto, Canada.

C126 Shanshan Jiang, Yuan Xue, and Douglas Schmidt, "Minimum Disruption Service Composition and Recovery in Mobile Ad hoc Networks, Proceedings of the 4th Annual International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MOBIQUITOUS 2007), August 6-10, 2007 - Philadelphia, PA.

C125 Jules White, Andrey Nechypurenko, Egon Wuchner, and Douglas C. Schmidt, "Optimizing and Automating Product-Line Variant Selection for Mobile Devices," Proceedings of the 11th International Software Product Line Conference, Kyoto, Japan, Sept 10-14, 2007.

C124 Nishanth Shankaran, Douglas C. Schmidt, Yingming Chen, Xenofon Koutsoukous, and Chenyang Lu, The Design and Performance of Configurable Component Middleware for End-to-End Adaptation of Distributed Real-time Embedded Systems, proceedings of the 10th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing, May 7-9, 2007, Santorini Island, Greece.

C123 Amogh Kavimandan, Krishnakumar Balasubramanian, Nishanth Shankaran, Aniruddha Gokhale, and Douglas C. Schmidt, QUICKER: A Model-driven QoS Mapping Tool, proceedings of the 10th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing (ISORC), May 7-9, 2007, Santorini Island, Greece.

C122 Krishnakumar Balasubramanian, Douglas C. Schmidt, Zoltan Molnar, and Akos Ledeczi, Component-based System Integration via (Meta)Model Composition, Proceedings of the 14th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS), March 26th-29th, 2007, Tucson, Arizona.

C121 Joe Hoffert, Shanshan Jiang, and Douglas C. Schmidt, "A Taxonomy of Discovery Services and Gap Analysis for Ultra-Large Scale Systems," Proceedings of the ACMSE 2007: 45th ACM Southeast Conference Winston-Salem, North Carolina, USA March 23-24, 2007.

C120 Andrey Nechypurenko, Egon Wuchner, Jules White, and Douglas C. Schmidt, Application of Aspect-based Modeling and Weaving for Complexity Reduction in Development of Automotive Distributed Realtime Embedded System, Proceedings of the Sixth ACM International Conference on Aspect-Oriented Software Development, Vancouver, British Columbia, March 12-16, 2007.

C119 Dipa Suri, Adam Howell, Douglas C. Schmidt, Gautam Biswas, John Kinnebrew, Will Otte, and Nishanth Shankaran, "A Multi-agent Architecture for Smart Sensing in the NASA Sensor Web," Proceedings of the 2007 IEEE Aerospace Conference Big Sky, Montana, March 3-10, 2007.

C118 John S. Kinnebrew, Ankit Gupta, Nishanth Shankaran, Gautam Biswas, and Douglas C. Schmidt, A Decision-Theoretic Planner with Dynamic Component Reconfiguration for Distributed Real-Time Applications, Proceedings of the The 8th International Symposium on Autonomous Decentralized Systems (ISADS 2007) Sedona, Arizona, Wednesday March 21 - Friday March 23, 2007.

C117 Ming Xiong, Jeff Parsons, James Edmondson, and Douglas C. Schmidt, "Evaluating Technologies for Tactical Information Management in Net-Centric Systems, Proceedings of the Defense Transformation and Net-Centric Systems conference, April 9-13, 2007, Orlando, Florida.

C116 Nilabja Roy, Nishanth Shankaran, and Douglas C. Schmidt "Target Manager: A Resource Provisioning Service for Enterprise Distributed Real-time and Embedded Systems," Proceedings of the International Symposium on Distributed Objects and Applications (DOA), Montpellier, France, Oct 29 - Nov 3, 2006.

C115 James Hill, John Slaby, Steve Baker, and Douglas C. Schmidt, "Evaluating Enterprise Distributed Real-time and Embedded System Quality of Service with System Execution Modeling Tools," Proceedings of the 12th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, Sydney, Australia, 16-18 August 2006.

C114 Nishanth Shankaran, Xenofon Koutsoukos, Chenyang Lu, Douglas C. Schmidt, and Yuan Xue, "Hierarchical Control of Multiple Resources in Distributed Real-time and Embedded Systems," Proceedings of the 18th Euromicro Conference on Real-Time Systems (ECRTS 06), Dresden, Germany, July 5-7, 2006.

C113 Dipa Suri, Adam Howell, Nishanth Shankaran, John Kinnebrew, Will Otte, Doug Schmidt, and Gautam Biswas, "Onboard Processing using the Adaptive Network Architecture", Proceedings of the Sixth annual NASA Earth Science Technology Conference MD, June 27th - 29th, 2006, College Park, MD.

C112 Gan Deng, Douglas C. Schmidt, Aniruddha Gokhale, and Andrey Nechypurenko, "Modularizing Variability and Scalability Concerns in Distributed Real-time and Embedded Systems with Modeling Tools and Component Middleware, Proceedings of the 9th IEEE International

Symposium on Object-oriented Real-time Distributed Computing (ISORC '06), April 24-26, 2006, Gyeongju, Korea.

C111 Stoyan Paunov and Douglas C. Schmidt, "RepoMan: A Component Repository Manager for Enterprise Distributed Real-time and Embedded Systems", Proceedings of the 44th ACM Southeast Conference, Melbourne, FL, March 10-12, 2006.

C110 Arvind S. Krishna, Aniruddha Gokhale, Douglas C. Schmidt, John Hatcliff, and Venkatesh Prasad Ranganat, "Context-Specific Middleware Specialization Techniques for Optimizing Software Product-line Architectures," Proceedings of ACM EuroSys 2006, Leuven, Belgium, April 18-21, 2006.

C109 Stoyan Paunov, James Hill, Douglas C. Schmidt, John Slaby, and Steve Baker, "Domain-Specific Modeling Languages for Configuring and Evaluating Enterprise DRE System Quality of Service," Proceedings of the 13th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS '06), March 27th-30th, 2006, University of Potsdam, Potsdam, Germany.

C108 Arvind S. Krishna, Aniruddha Gokhale, Douglas C. Schmidt, John Hatcliff, and Venkatesh Prasad Ranganat, "Towards Highly Optimized Real-time Middleware for Software Product-line Architectures," Proceedings of the Work-In-Progress session at the 26th IEEE Real-Time Systems Symposium, December 5-8, 2005, Miami, Florida.

C107 Gan Deng, Jaiganesh Balasubramanian, William Otte, Douglas C. Schmidt, and Aniruddha Gokhale, DAnCE: A QoS-enabled Component Deployment and Configuration Engine, Proceedings of the 3rd Working Conference on Component Deployment, Grenoble, France, November 28-29, 2005, p. 67-82.

C106 Jaiganesh Balasubramanian, Balachandran Natarajan, Douglas C. Schmidt, Aniruddha Gokhale, Gan Deng, and Jeff Parsons, "Evaluating Techniques for Dynamic Component Updating," Proceedings of the International Symposium on Distributed Objects and Applications (DOA), Agia Napa, Cyprus, Oct 31 - Nov 4, 2005, p. 978-996.

C105 Jules White, Douglas Schmidt, and Aniruddha Gokhale, "Simplifying Autonomic Enterprise Java Bean Applications via Model-driven Development: a Case Study," Proceedings of MODELS 2005, ACM/IEEE 8th International Conference on Model Driven Engineering Languages and Systems, Half Moon Resort, Montego Bay, Jamaica, October 5-7, 2005, p. 601-615.

C104 Arvind Krishna, Douglas C. Schmidt, and Michael Stal, "Context Object A Design Pattern for Efficient Middleware Request Processing," Proceedings of the 12th Pattern Language of Programming Conference, Allerton Park, Illinois, September 7-10, 2005.

C103 Michael Stal and Douglas C. Schmidt, "Activator," Proceedings of the 12th Pattern Language of Programming Conference, Allerton Park, Illinois, September 7-10, 2005.

C102 Jules White, Boris Kolpackov, Balachandran Natarajan, and Douglas C. Schmidt, "Reducing Code Complexity With Vocabulary-Specific XML Language Bindings," Proceedings of the 43nd ACM Southeastern conference in Atlanta, GA, March 2005.

C101 Cemal Yilmaz, Arvind Krishna, Atif Memon, Adam Porter, Douglas C. Schmidt, Aniruddha Gokhale, and Bala Natarajan, "A Distributed Continuous Quality Assurance Process for Monitoring Performance Degradation in Evolving Software Systems, proceedings of the 27th International Conference on Software Engineering, St. Louis, MO, May 15-21, 2005, p. 293-302.

C100 Arvind S. Krishna, Emre Turkay, Aniruddha Gokhale, and Douglas C. Schmidt, Model-Driven Techniques for Evaluating the QoS of Middleware Configurations for DRE Systems, Proceedings of the 11th IEEE Real-Time and Embedded Technology and Applications Symposium, San Francisco, CA, March 2005, p. 180-189.

C99 Krishnakumar Balasubramanian, Jaiganesh Balasubramanian, Jeff Parsons, Aniruddha Gokhale, and Douglas C. Schmidt, "A Platform-Independent Component Modeling Language for Distributed Real-time and Embedded Systems," Proceedings of the 11th IEEE Real-Time and Embedded Technology and Applications Symposium, San Francisco, CA, March 2005, p. 190-199.

C98 Nanbor Wang, Christopher Gill, Douglas C. Schmidt, and Venkita Subramonian, "Configuring Real-time Aspects in Component Middleware," Proceedings of the Conference on Distributed Objects and Applications (DOA 2004), October 25-29, 2004, Cyprus, Greece.

C97 Jaiganesh Balasubramanian, Douglas C. Schmidt, Lawrence Dowdy, and Ossama Othman, "Evaluating the Performance of Middleware Load Balancing Strategies," Proceedings of the 8th International IEEE Enterprise Distributed Object Computing Conference, Monterey, California, September 20-24, 2004.

C96 George Edwards, Gan Deng, Douglas C. Schmidt, Aniruddha Gokhale, and Balachandran Natarajan, "Model-driven Configuration and Deployment of Component Middleware Publisher/Subscriber Services," Proceedings of the 3rd ACM International Conference on Generative Programming and Component Engineering, Vancouver, CA, October 2004.

C95 Andrey Nechypurenko, Douglas C. Schmidt, Tao Lu, Gan Deng, Emre Turkey, and Aniruddha Gokhale, "Concern-based Composition and Reuse of Distributed Systems," Proceedings of the 8th International Conference on Software Reuse, ACM/IEEE, Madrid, Spain, July 2004.

C94 Arvind Krishna, Douglas C. Schmidt, Adam Porter, Atif Memon, Diego Sevilla-Ruiz, "Improving the Quality of Performance-intensive Software via Model-

integrated Distributed Continuous Quality Assurance," Proceedings of the 8th International Conference on Software Reuse, ACM/IEEE, Madrid, Spain, July 2004.

C93 Chris Gill, Jeanna M. Gossett, David Corman, Joseph P. Loyall, Richard E. Schantz, Michael Atighetchi, and Douglas C. Schmidt, "Integrated Adaptive QoS Management in Middleware: An Empirical Case Study," Proceedings of the 10th Real-time Technology and Application Symposium, May 25-28, 2004, Toronto, CA.

C92 Pradeep Gore, Douglas C. Schmidt, Chris Gill, and Irfan Pyarali, "The Design and Performance of a Real-time Notification Service," Proceedings of the 10th IEEE Real-time Technology and Application Symposium (RTAS '04), Toronto, CA, May 2004.

C91 Yamuna Krishnamurthy, Chris Gill, Douglas C. Schmidt, Irfan Pyarali, Louis Mgeta, Yuanfang Zhang, and Stephen Torri, "The Design and Performance of Real-time CORBA 2.0: Dynamic Scheduling in TAO," Proceedings of the 10th IEEE Real-time Technology and Application Symposium (RTAS '04), Toronto, CA, May 2004.

C90 Arvind S. Krishna, Nanbor Wang, Balachandran Natarajan, Aniruddha Gokhale, Douglas C. Schmidt and Gautam Thaker, "CCMPerf: A Benchmarking Tool for CORBA Component Model Implementations," Proceedings of the 10th IEEE Real-time Technology and Application Symposium (RTAS '04), Toronto, CA, May 2004.

C89 George Edwards, Douglas C. Schmidt, Aniruddha Gokhale, and Bala Natarajan, Integrating Publisher/Subscriber Services in Component Middleware for Distributed Real-time and Embedded Systems, Proceedings of the 42nd ACM Southeastern conference in Huntsville, AL, April 2004.

C88 Atif Memon, Adam Porter, Cemal Yilmaz, Adithya Nagarajan, Douglas C. Schmidt, and Bala Natarajan, "Skoll: Distributed Continuous Quality Assurance," Proceedings of the 26th IEEE/ACM International Conference on Software Engineering, Edinburgh, Scotland, May 2004.

C87 Arvind Krishna, Douglas C. Schmidt, and Raymond Klefstad, "Enhancing Real-Time CORBA via Real-Time Java," Proceedings of the 24th IEEE International Conference on Distributed Computing Systems (ICDCS), March 23-26, 2004, Tokyo, Japan.

C86 Arvind Krishna, Douglas C. Schmidt, Krishna Raman, and Raymond Klefstad, "Enhancing Real-time CORBA Predictability and Performance," Proceedings of the 5th International Symposium on Distributed Objects and Applications (DOA), Catania, Sicily, November 2003.

C85 Jeff Gray, Ted Bapty, Sandeep Neema, Douglas C. Schmidt, Aniruddha Gokhale, and Balachandran Natarajan, "An Approach for Supporting Aspect-Oriented Domain Modeling, Proceedings of the 2nd Generative Programming and Component Engineering (GPCE '03) conference, Erfurt, Germany, September 22-25, 2003.

C84 Richard E. Schantz, Joseph P. Loyall, Douglas C. Schmidt, Craig Rodrigues, Yamuna Krishnamurthy, and Irfan Pyarali, "Flexible and Adaptive QoS Control for Distributed Real-time and Embedded Middleware," Proceedings of Middleware 2003, 4th IFIP/ACM/USENIX International Conference on Distributed Systems Platforms, June 16-20, 2003, Rio de Janeiro, Brazil.

C83 Arvind Krishna, Douglas C. Schmidt, Raymond Klefstad, and Angelo Corsaro, "Towards Predictable Real-time Java Object Request Brokers," Proceedings of the 9th IEEE Real-time/Embedded Technology and Applications Symposium (RTAS), Washington DC, May 28-30, 2003.

C82 Douglas C. Schmidt and Frank Buschmann, "Patterns, Frameworks, and Middleware: Their Synergistic Relationships," Proceedings of the IEEE/ACM International Conference on Software Engineering, Portland, Oregon, May 3–10, 2003.

C81 Radu Cornea, Nikil Dutt, Rajesh Gupta, Ingolf Krueger, Alex Nicolau, Douglas C. Schmidt, and Sandeep Shukla, "FORGE: A Framework for Optimization of Distributed Embedded Systems Software," International Parallel and Distributed Processing Symposium, Nice, France, April 22-26 2003.

C80 Raymond Klefstad, Sumita Rao, and Douglas C. Schmidt, "Design and Performance of a Dynamically Configurable, Messaging Protocols Framework for Real-time CORBA," Proceedings of the Distributed Object and Component-based Software Systems part of the Software Technology Track at the 36th Annual Hawaii International Conference on System Sciences, January 6 – 9, 2003, Big Island of Hawaii.

C79 Raymond Klefstad, Arvind S. Krishna, and Douglas C. Schmidt, "Design and Performance of a Modular Portable Object Adapter for Distributed, Real-Time, Embedded CORBA Applications," Proceedings of the Distributed Objects and Applications (DOA) conference, Irvine, CA, October/November, 2002.

C78 Chris Gill, Fred Kuhns, Douglas C. Schmidt, and Ron Cytron, "Empirical Differences Between COTS Middleware Scheduling Paradigms," Proceedings of the Distributed Objects and Applications (DOA) conference, Irvine, CA, October/November, 2002.

C77 Angelo Corsaro and Douglas C. Schmidt, "jRate: The Chameleonic Real-Time Java Implementation," Proceedings of the Distributed Objects and Applications (DOA) conference, Irvine, CA, October/November, 2002.

C76 Mayur Deshpande, Douglas C. Schmidt, Carlos O'Ryan, and Darrell Brunsch, "The Design and Performance of Asynchronous Method Handling for CORBA," Proceedings of the Distributed Objects and Applications (DOA) conference, Irvine, CA, October/November, 2002.

C75 Irfan Pyarali, Douglas C. Schmidt, and Ron Cytron, "Achieving End-to-End Predictability of the TAO Real-time CORBA ORB," Proceedings of the 8^{th} IEEE Real-Time Technology and Applications Symposium, San Jose, CA, September 2002.

C74 Angelo Corsaro and Douglas C. Schmidt, "Evaluating Real-Time Java Features and Performance for Real-time Embedded Systems," Proceedings of the 8th IEEE Real-Time Technology and Applications Symposium, San Jose, CA, September 2002.

C73 Angelo Corsaro, Douglas C. Schmidt, Raymond Klefstad, and Carlos O'Ryan, "Virtual Component: a Design Pattern for Memory-Constrained Embedded Applications," Proceedings of the 9th Annual Conference on the Pattern Languages of Programs, Monticello, Illinois, September, 2002.

c72 Joseph K. Cross and Douglas C. Schmidt, "Quality Connector: A Pattern Language for Provisioning and Managing Quality-Constrained Services in Distributed Real-time and Embedded Systems Proceedings of the 9th Annual Conference on the Pattern Languages of Programs, Monticello, Illinois, September, 2002.

C71 Richard Schantz, Franklin Webber, Partha Pal, Joseph Loyall, and Douglas C. Schmidt, "Protecting Applications Against Malice with Adaptive Middleware," Certification and Security in E-Services stream of the 17th IFIP World Computer Congress, Montreal, Canada, August 25-30, 2002.

C70 Richard Schantz and Douglas C. Schmidt, "Research Advances in Middleware for Distributed Systems: State of the Art," Computer Communications stream of the 17th IFIP World Computer Congress, Montreal, Canada, August 25-30, 2002.

C69 Raymond Klefstad, Douglas C. Schmidt, and Carlos O'Ryan, "Towards Highly Configurable Real-time Object Request Brokers," the IEEE International Symposium on Object-Oriented Real-time Distributed Computing (ISORC), Washington DC, April 29 – May 1, 2002.

C68 Angelo Corsaro, Douglas C. Schmidt, Chris Gill, and Ron Cytron, "Formalizing Meta-Programming Techniques to Reconcile Heterogeneous Scheduling Policies in Open Distributed Real-Time Systems," Proceedings of the 3rd International Symposium on Distributed Objects and Applications, September 8-10, 2001, Rome, Italy.

C67 David A. Karr, Craig Rodrigues, Yamuna Krishnamurthy, Irfan Pyarali, and Douglas C. Schmidt "Application of the QuO Quality-of-Service Framework to a Distributed Video Application," Proceedings of the 3rd International Symposium on Distributed Objects and Applications, September 8-10, 2001, Rome, Italy.

C66 Nanbor Wang, Kirthika Parameswaran, and Douglas C. Schmidt, "The Design and Performance of Meta-Programming Mechanisms for Object Request Broker Middleware," Proceedings of the 6th USENIX Conference on Object-Oriented Technologies and Systems (COOTS), San Antonio, TX, Jan/Feb, 2001.

C65 Andy Gokhale, Bala Natarajan, Douglas C. Schmidt and Shalini Yajnik, "Applying Patterns to Improve the Performance of Fault-Tolerant CORBA," of the 7th International Conference on High Performance Computing (HiPC 2000), ACM/IEEE, Bangalore, India, December 2000.

C64 Nanbor Wang, Michael Kircher, and Douglas C. Schmidt, "Applying Reflective Techniques to Optimize a QoS-enabled CORBA Component Model Implementation, the 24th Annual International Computer Software and Applications Conference (COMPSAC 2000), Taipai, Taiwan, October 25-27 2000.

C63 Frederic Andres, Nicolas Dessaigne, Jose Martinez, Noureddine Mouaddib, Kinji Ono, Douglas C. Schmidt, Panrit Tosukhowong, "MISE: The MediaSys Image Search Engine," 11th International Conference on Database and Expert Systems Applications (DEXA 2000), London, UK, September 2000.

C62 Andy Gokhale, Bala Natarajan, Douglas C. Schmidt and Shalini Yajnik, "DOORS: Towards High-performance Fault-Tolerant CORBA," Proceedings of the 2nd International Symposium on Distributed Objects and Applications (DOA '00), OMG, Antwerp, Belgium, September 2000.

C61 Irfan Pyarali, Carlos O'Ryan, Douglas C. Schmidt, "A Pattern Language for Efficient, Predictable, Scalable, and Flexible Dispatching Components," Proceedings of the 7th Pattern Language of Programming Conference, Monticello, Illinois, August, 2000.

C60 Douglas C. Schmidt, Carlos O'Ryan, Irfan Pyarali, Michael Kircher and Frank Buschmann, "Leader/Followers: A Design Pattern for Efficient Multi-threaded Event Demultiplexing and Dispatching," Proceedings of the 7th Pattern Languages of Programming Conference, Monticello, Illinois, August 2000.

C59 Carlos O'Ryan, Douglas C. Schmidt, Fred Kuhns, Marina Spivak, Jeff Parsons Irfan Pyarali, and David L. Levine, "Evaluating Policies and Mechanisms for Supporting Embedded, Real-Time Applications with CORBA 3.0," Proceedings of the Sixth IEEE Real-Time Technology and Applications Symposium (RTAS'00), Washington D.C., USA, May 31-June 2, 2000.

C58 Nanbor Wang, Douglas C. Schmidt, and David Levine, "Optimizing the CORBA Component Model for High-performance and Real-time Applications," Work-in-progress session of the IFIP/ACM Middleware 2000 Conference, Pallisades, New York, April 3-7, 2000.

C57 Alexander B. Arulanthu, Carlos O'Ryan, Douglas C. Schmidt, Michael Kircher, and Jeff Parsons, "The Design and Performance of a Scalable ORB Architecture for CORBA Asynchronous Messaging," Proceedings of the IFIP/ACM Middleware 2000 Conference, Pallisades, New York, April 3-7, 2000.

C56 Carlos O'Ryan, Fred Kuhns, Douglas C. Schmidt, Ossama Othman, and Jeff Parsons, The Design and Performance of a Pluggable Protocols Framework for Real-time Distributed Object Computing Middleware, Proceedings of the IFIP/ACM Middleware 2000 Conference, Pallisades, New York, April 3-7, 2000.

C55 Irfan Pyarali, Carlos O'Ryan, and Douglas C. Schmidt, "A Pattern Language for Efficient, Predictable, Scalable, and Flexible Dispatching Mechanisms for Distributed Object Computing Middleware," Proceedings of the IEEE/IFIP

International Symposium on Object-Oriented Real-time Distributed Computing, March 15-17, 2000, Newport Beach, California.

C54 David Levine, Douglas C. Schmidt, and Sergio Flores-Gaitan, "An Empirical Evaluation of OS Support for Real-time CORBA Object Request Brokers," Proceedings of the Multimedia Computing and Networking 2000 (MMCN00) conference, ACM, San Jose, CA, January 25-27 2000.

C53 Douglas C. Schmidt, "Middleware Techniques and Optimizations for Real-time, Embedded Systems," Proceedings of the 12th International Symposium On System Synthesis, IEEE, San Jose, CA, USA November, 11, 1999.

C52 Panrit Tosukhowong, Frederic Andres, Kinji Ono, Nicolas Dessaigne, Josi Martinez, Nouredine Mouaddib, Douglas C. Schmidt, "A Flexible Image Search Engine," Proceedings of International Multimedia Conference Archive Proceedings of the Seventh ACM International Conference on Multimedia, Orlando, Florida, United States, November 1-5, 1999.

C51 Bryan S. Doerr, Thomas Venturella, Rakesh Jha, Christopher D. Gill, and Douglas C. Schmidt, "Adaptive Scheduling for Real-time, Embedded Information Systems," Proceedings of the 18th IEEE/AIAA Digital Avionics Systems Conference (DASC), St. Louis, Missouri, October 24-29, 1999.

C50 Christopher D. Gill, David L. Levine, Carlos O'Ryan, and Douglas C. Schmidt, "Distributed Object Visualization for Sensor-Driven Systems," Proceedings of the 18th IEEE/AIAA Digital Avionics Systems Conference (DASC), St. Louis, Missouri, October 24-29, 1999.

C49 Fred Kuhns, Douglas C. Schmidt, and David L. Levine, "The Performance of a Real-time I/O Subsystem for QoS-enabled ORB Middleware," Proceedings of the International Symposium on Distributed Objects and Applications (DOA '99), Edinburgh, Scotland, September 1999.

C48 David L. Levine, Christopher D. Gill, and Douglas C. Schmidt, "Object Lifecycle Manager – A Complementary Pattern for Controlling Object Creation and Destruction." Proceedings of the 5th Pattern Languages of Programing Conference, Allerton Park, Illinois, USA, 15 – 18 August 1999.

C47 Fred Kuhns, Douglas C. Schmidt, David Levine, and Rajeev Bector, "The Design and Performance of a Real-time I/O Subsystem," Proceedings of the 5th IEEE Real-Time Technology and Applications Symposium (RTAS99), Vancouver, British Columbia, Canada, June 2-4, 1999.

C46 Irfan Pyarali, Carlos O'Ryan, Douglas C. Schmidt, Nanbor Wang, Vishal Kachroo, and Aniruddha Gokhale, "Applying Optimization Patterns to Design Real-time ORBs," Proceedings of the 5th USENIX Conference on Object-Oriented Technologies and Systems, May 3-7, 1999, San Diego, CA.

C45 Andy Gokhale and Douglas C. Schmidt, "Techniques for Optimizing CORBA Middleware for Distributed Embedded Systems" Proceedings of INFOCOM '99, March 21-25th, New York, New York.

C44 Sumedh Mungee, Nagarajan Surendran, and Douglas C. Schmidt, "The Design and Performance of a CORBA Audio/Video Streaming Service,"

Proceedings of the 31st Hawaii International Conference on System Systems (HICSS), Hawaii, January, 1999, minitrack on Multimedia DBMS and the WWW, Hawaii, January 1999.

C43 Chris D. Gill, David L. Levine, and Douglas C. Schmidt, "Dynamic Scheduling for Avionics Applications," Proceedings of the 17th IEEE/AIAA Digital Avionics System Conference, 31 October - 6 November 1998.

C42 James Hu, Irfan Pyarali, and Douglas C. Schmidt, "Applying the Proactor Pattern to High-Performance Web Servers," Proceedings of the 10th International Conference on Parallel and Distributed Computing and Systems, IASTED, Las Vegas, Nevada, October 28-31, 1998.

C41 Douglas C. Schmidt, Sumedh Mungee, and Andy Gokhale, "Alleviating Priority Inversion and Non-determinism in Real-time CORBA ORB Core Architectures," Proceedings of the Fourth IEEE Real-Time Technology and Applications Symposium (RTAS), Denver, Colorado, June 3-5, 1998.

C40 Prashant Jain, Seth Widoff, and Douglas C. Schmidt, "The Design and Performance of MedJava, A Distributed Electronic Medical Imaging System Developed with Java Applets and Web Tools" Proceedings of the 4th USENIX Conference on Object-Oriented Technologies and Systems, Sante Fe, New Mexico, April 1998. This was selected as the best student paper in the conference.

C39 James Hu, Sumedh Mungee, and Douglas C. Schmidt, "Techniques for Developing and Measuring High-performance Web Servers over ATM Networks," Proceedings of INFOCOM '98, San Francisco, March/April, 1998.

C38 Aniruddha Gokhale and Douglas C. Schmidt, "Optimizing the Performance of the CORBA Internet Inter-ORB Protocol Over ATM," Proceedings of the 31st Hawaii International Conference on System Systems (HICSS), Hawaii, January, 1998. This was selected as the best paper in the Software Technology Track (188 submitted, 77 accepted).

C37 Aniruddha Gokhale and Douglas C. Schmidt, "Evaluating the Performance of Demultiplexing Strategies for Real-time CORBA," Proceedings of GLOBECOM '97 conference, IEEE, Phoenix, AZ, November, 1997.

C36 James Hu, Irfan Pyarali, and Douglas C. Schmidt, "Measuring the Impact of Event Dispatching and Concurrency Models on Web Server Performance Over High-speed Networks," Proceedings of the 2nd Global Internet Conference (held as part of GLOBECOM '97) in Phoenix, AZ, November 4-8, 1997.

C35 Tim Harrison and David Levine and Douglas C. Schmidt, "The Design and Performance of a Real-time CORBA Event Service," Proceedings of OOPSLA '97, ACM, Atlanta, GA, October 1997.

C34 Aniruddha Gokhale and Douglas C. Schmidt and Stan Moyer, "Tools for Automating the Migration from DCE to CORBA,"Proceedings of ISS 97: World Telecommunications Congress, IEEE Toronto, Canada, September, 1997.

C33 Douglas C. Schmidt, Tim H. Harrison, and Nat Pryce, "Thread-specific Storage: an Object Behavioral Pattern for Efficiently Accessing per-Thread State," The 4th annual Pattern Languages of Programming conference in Allerton Park, Illinois, September 1997.

C32 Irfan Pyarali, Tim Harrison, Douglas C. Schmidt, and Thomas Jordan, "Proactor: an Object Behavioral Pattern for Demultiplexing and Dispatching Handlers for Asynchronous Events," the 4th annual Pattern Languages of Programming conference in Allerton Park, Illinois, September 1997.

C31 Prashant Jain and Douglas C. Schmidt, "Service Configurator – A Pattern for Dynamic Configuration of Services," Proceedings of the 3rd Conference on Object-Oriented Technologies and Systems, USENIX, Portland, OR, June 16-19, 1997.

C30 Aniruddha Gokhale and Douglas C. Schmidt, "Evaluating Latency and Scalability of CORBA Over High-Speed ATM Networks," Proceedings of the International Conference on Distributed Computing Systems '97, IEEE, Baltimore, Maryland, May 27–30, 1997.

C29 Aniruddha Gokhale and Douglas C. Schmidt, "Performance of the CORBA Dynamic Invocation Interface and Internet Inter-ORB Protocol over High-Speed ATM Networks," Proceedings of GLOBECOM '96, IEEE, London England, November, 1996.

C28 Aniruddha Gokhale and Douglas C. Schmidt, "Measuring the Performance of Communication Middleware on High-Speed Networks," Proceedings of SIGCOMM '96, ACM, San Francisco, August 28-30th, 1996.

C27 Irfan Pyarali, Tim Harrison, and Douglas. C. Schmidt, "Design and Performance of an Object-Oriented Framework for High-Speed Electronic Medical Imaging," Proceedings of the 2nd Conference on Object-Oriented Technologies and Systems (COOTS), USENIX, Toronto, June 18-22, 1996.

C26 Douglas C. Schmidt, "A Family of Design Patterns For Flexibly Configuring Network Services in Distributed Systems," Proceedings of the International Conference on Configurable Distributed Systems, IEEE, Annapolis, Maryland, May 6-8, 1996.

C25 Douglas C. Schmidt "Using Design Patterns to Develop High-Performance Object-Oriented Communication Software Frameworks," Proceedings of the 8th Annual Software Technology Conference, Salt Lake City, Utah, April 21-26, 1996.

C24 Douglas C. Schmidt, Timothy H. Harrison, and Irfan Pyarali, "An Object-Oriented Framework for High-Performance Electronic Medical Imaging," Proceedings of the *Very High Resolution and Quality Imaging* mini-conference at the Symposium on Electronic Imaging in the International Symposia Photonics West 1996, SPIE, San Jose, California USA, January 27 - February 2, 1996.

C23 Douglas C. Schmidt and Charles D. Cranor, "Half-Sync/Half-Async: A Pattern for Efficient and Well-structured Concurrent I/O," *The 2nd Pattern*

Languages of Programs Conference Monticello, Illinois, September 6-8, 1995.

C22 R. Greg Lavender and Douglas C. Schmidt, "Active Object: An Object Behavioral Pattern for Concurrent Programming," *The 2nd Pattern Languages of Programs Conference*, Monticello, Illinois, September 6-8, 1995.

C21 Guru Parulkar, Douglas C. Schmidt, and Jonathan S. Turner, "*aItPm*: a Strategy for Integrating IP with ATM," the Symposium on Communications Architectures and Protocols (SIGCOMM), ACM, Cambridge, MA, August 30 to September 1, 1995.

C20 Douglas C. Schmidt, Tim Harrison, and Ehab Al-Shaer, "Object-Oriented Components for High-speed Network Programming," *Proceedings of the Conference on Object-Oriented Technologies (COOTS)*, USENIX, June 26-29, 1995 Monterey, California, USA, pp. 21–38.

C19 Douglas C. Schmidt and Paul Stephenson, "Experience Using Design Patterns to Evolve Communication Software Across Diverse OS Platforms," *Proceedings of the 9th European Conference on Object-Oriented Programming (ECOOP)*, ACM, Aarhus, Denmark, August, 1995.

C18 Douglas C. Schmidt and Tatsuya Suda, "Measuring the Performance of Parallel Message-based Process Architectures," *Proceedings of the INFOCOM Conference on Computer Communications*, IEEE, Boston, MA, April, 1995, pp. 624–633.

C17 Douglas C. Schmidt and Tatsuya Suda, "Experiences with an Object-Oriented Architecture for Developing Dynamically Extensible Network Management Software," *Proceedings of the Globecom Conference*, IEEE, San Francisco, California, November, 1994, pp. 1–7.

C16 Douglas C. Schmidt and Paul Stephenson, "Achieving Reuse Through Design Patterns," *Proceedings of the 3rd Annual C++ World Conference*, SIGS, Austin, Texas, November 14-18, 1994.

C15 Douglas C. Schmidt, "Developing Object-Oriented Frameworks to Dynamically Configure Concurrent, Multi-service Network Daemons," *Proceedings of the 3rd Annual C++ World Conference*, SIGS, Austin, Texas, November 14-18, 1994.

C14 Douglas C. Schmidt, "Reactor: An Object Behavioral Pattern for Concurrent Event Demultiplexing and Dispatching," *The 1st Annual Conference on the Pattern Languages of Programs*, Monticello, Illinois, August, 1994, pp. 1–10.

C13 Douglas C. Schmidt, "The ADAPTIVE Communication Environment: An Object-Oriented Network Programming Toolkit for Developing Communication Software," *Proceedings of the 12th Annual Sun Users Group Conference*, SUG, San Francisco, June 16-17, 1994. This paper won the "best student paper" award at the conference.

C12 Douglas C. Schmidt, Burkhard Stiller, Tatsuya Suda, and Martina Zitterbart, "Configuring Function-based Communication Protocols for Distributed Applications," *Proceedings of the 8th International Working Conference on*

Upper Layer Protocols, Architectures, and Applications, IFIP, Barcelona, Spain, June 1-3, 1994, pp. 361–376.

C11 Douglas C. Schmidt and Tatsuya Suda, "The ADAPTIVE Service Executive: An Object-Oriented Architecture for Configuring Concurrent Distributed Communication Systems," *Proceedings of the 8th International Working Conference on Upper Layer Protocols, Architectures, and Applications*, IFIP, Barcelona, Spain, June 1-3, 1994, pp. 163–178.

C10 Douglas C. Schmidt, "ASX: An Object-Oriented Framework for Developing Distributed Applications," *Proceedings of the 6th C++ Conference*, USENIX, Cambridge, Massachusetts, April, 1994, pp. 200–220.

C9 Douglas C. Schmidt, "The ADAPTIVE Communication Environment: Object-Oriented Network Programming Components for Developing Client/Server Applications," *Proceedings of the 11th Annual Sun Users Group Conference*, SUG, San Jose, December 7-9, 1993, pp. 214–225. This paper won the "best student paper" award at the conference.

C8 Douglas C. Schmidt and Paul Stephenson, "An Object-Oriented Framework for Developing Network Server Daemons," *Proceedings of the 2nd Annual C++ World Conference*, SIGS, Dallas, Texas, October 18-22, 1993, pp. 73–85.

C7 Douglas C. Schmidt, "Object-Oriented Techniques for Developing Extensible Network Servers," *Proceedings of the 2nd Annual C++ World Conference*, SIGS, Dallas, Texas, October 18-22, 1993.

C6 Douglas C. Schmidt, Burkhard Stiller, Tatsuya Suda, Ahmed Tantawy, and Martina Zitterbart, "Configuration Support for Flexible Function-Based Communication Systems," *Proceedings of the 18th Conference on Local Computer Networks*, IEEE, Minneapolis, Minnesota, September 20-22, 1993, pp. 369–378.

C5 Douglas C. Schmidt and Tatsuya Suda, "ADAPTIVE: a Framework for Experimenting with High-Performance Transport System Process Architectures," *Proceedings of the 2nd International Conference on Computer Communications and Networks*, ISCA, San Diego, California, June 28-30, 1993, pp. 1–8.

C4 Donald F. Box, Douglas C. Schmidt, and Tatsuya Suda, "ADAPTIVE: An Object-Oriented Framework for Flexible and Adaptive Communication Protocols," *Proceedings of the 4th Conference on High Performance Networking*, IFIP, Liege, Belgium, December 14-18, 1992, pp. 367–382.

C3 Douglas C. Schmidt, Donald F. Box, and Tatsuya Suda, "ADAPTIVE: A Flexible and Adaptive Transport System Architecture to Support Lightweight Protocols for Multimedia Applications on High-Speed Networks," *Proceedings of the 1st Symposium on High Performance Distributed Computing*, IEEE, Syracuse, New York, September 9-11, 1992, pp. 174–186.

C2 Richard W. Selby, Adam A. Porter, Douglas C. Schmidt, and James Berney, "Metric-Driven Analysis and Feedback Systems for Enabling Empirically Guided Software Development," *Proceedings of the 13th Annual*

International Conference on Software Engineering, IEEE, Austin, Texas, May, 1991, pp. 430–443.

C1 Douglas C. Schmidt "GPERF: A Perfect Hash Function Generator," *Proceedings of the 2nd C++ Conference*, USENIX, San Francisco, California, April 9-11, 1990, pp. 87–102.

- **Refereed Workshop Publications**

W70 Violetta Vylegzhanina, Douglas C. Schmidt, Pamela Hull, Janice S. Emerson, Meghan E. Quirk, and Shelagh Mulvaney, "Helping Children Eat Well Via Mobile Software Technologies," Proceedings of the Second International Workshop on Mobile Development Lifecycle, October 21st, 2015, Portland, OR.

W69 Jules White and Douglas C. Schmidt, "R&D Challenges and Emering Softwares for Multicore Deployment/Configuration Optimization," proceedings of the ACM Workshop on Future of Software Engineering Research (FoSER 2010), Santa Fe, NM, November 7-11, 2010.

W68 Will Otte, Douglas C. Schmidt, and Aniruddha Gokhale, "Towards an Adaptive Deployment and Configuration Framework for Component-based Distributed Systems," Proceedings of the 9th Workshop on Adaptive and Reflective Middleware (ARM 2010) November 27, 2010, Bangalore India, collocated with Middleware 2010.

W67 Jaiganesh Balasubramanian, Alexander Mintz, Andrew Kaplan, Grigory Vilkov, Artem Gleyzer, Antony Kaplan, Ron Guida, Pooja Varshneya and Douglas Schmidt, "Adaptive Parallel Computing for Large-scale Distributed and Parallel Applications," Proceedings of the Workshop on Data Dissemination for Large-scale Complex Critical Infrastructures (DD4LCCI), 27 April 2010, in conjunction with EDCC 2010, Valencia - Spain, April 28-30, 2010.

W66 Joe Hoffert, Douglas Schmidt, and Aniruddha Gokhale, "Adapting and Evaluating Distributed Real-time and Embedded Systems in Dynamic Environments," Proceedings of the Workshop on Data Dissemination for Large-scale Complex Critical Infrastructures (DD4LCCI), 27 April 2010, in conjunction with EDCC 2010, Valencia - Spain, April 28-30, 2010.

W65 Joe Hoffert, Dan Mack, and Douglas C. Schmidt, "Using Machine Learning to Maintain Pub/Sub System QoS in Dynamic Environments, Proceedings of the 8th Workshop on Adaptive and Reflective Middleware (ARM'09) December 1st 2009, Urbana Champaign, Illinois, USA collocated with Middleware 2009.

W64 Chris Thompson, Jules White, Brian Dougherty, and Douglas C. Schmidt, "Optimizing Mobile Application Performance with Model-Driven Engineering," Proceedings of the 7th IFIP Workshop on Software Technologies for Future Embedded and Ubiquitous Systems (SEUS 2009), November 16-18, 2009, Newport Beach, California.

W63 Jules White and Douglas C. Schmidt, "Filtered Cartesian Flattening: An Approximation Technique for Optimally Selecting Features while Adhering to Resource Constraints," proceedings of the Workshop on Analyses of

Software Product Lines (ASPL 2008) at the Software Product Lines Conference (SPLC), September 8-12, 2008, Limerick, Ireland.

W62 Joe Hoffert, Douglas C. Schmidt, Mahesh Balakrishnan, and Ken Birman, Supporting Large-scale Continuous Stream Datacenters via Pub/Sub Middleware and Adaptive Transport Protocols, Proceedings of the 2nd workshop on Large-Scale Distributed Systems and Middleware (LADIS 2008), IBM TJ Watson Research Center, Yorktown, New York, September 2008.

W61 Nishanth Shankaran, John S. Kinnebrew, Xenofon D. Koutsoukos, Chenyang Lu, Douglas C. Schmidt, and Gautam Biswas, Towards an Integrated Planning and Adaptive Resource Management Architecture for Distributed Real-time Embedded Systems," Proceedings of the Workshop on Adaptive and Reconfigurable Embedded Systems (APRES) at the 14th IEEE Real-Time and Embedded Technology and Applications Symposium, St. Louis, MO, United States, April 22 - April 24, 2008.

W60 Serena Fritsch, Aline Senart, Douglas C. Schmidt, and Siobhan Clarke, "Scheduling Time-bounded Dynamic Software Adaptation," Proceedings of the workshop on Software Engineering for Adaptive and Self-Managing Systems at the 30th IEEE/ACM International Conference on Software Engineering May 12-13, 2008, Leipzig, Germany.

W59 James Hill, Jules White, Sean Eade, and Douglas C. Schmidt, "Towards a Solution for Synchronizing Disparate Models of Ultra-Large-Scale Systems," Proceedings of the Second International Workshop on Ultra-Large-Scale Software- Intensive Systems at the 30th IEEE/ACM International Conference on Software Engineering May 10-11, 2008, Leipzig, Germany.

W58 Douglas C. Schmidt and Hans van't Hag, "Addressing the Challenges of Tactical Information Management in Net-Centric Systems with OpenSplice DDS," Proceedings of the 16th International ACM Workshop on Parallel and Distributed Real-Time Systems (WPDRTS '08), Miami, Florida, April 2008.

W57 Shanshan Jiang, Yuan Xue, and Douglas C. Schmidt, "Disruption-Aware Service Composition and Recovery in Dynamic Networking Environments," WoRkshop on Automating Service Quality (WRAQS) 2007, Co-Located with ASE 2007 November 6, 2007, Atlanta, Georgia.

W56 Jules White, Douglas C. Schmidt, Sean Mulligan, "The Generic Eclipse Modeling System," Model-Driven Development Tool Implementer's Forum, TOOLS '07, June, 2007, Zurich, Switzerland.

W55 John S. Kinnebrew, Nishanth Shankaran, Gautam Biswas, and Douglas C. Schmidt, A Decision-Theoretic Planner with Dynamic Component Reconfiguration for Distributed Real-time and Embedded Systems, Proceedings of the Workshop on Artificial Intelligence for Space Applications at IJCAI 2007, January 6-12, 2007, Hyderabad, India.

W54 Andrey Nechypurenko, Jules White, Egon Wuchner, and Douglas C. Schmidt, "Applying Model Intelligence Frameworks for Deployment Problem in Real-time and Embedded Systems," Proceedings of MARTES: Modeling and

Analysis of Real-Time and Embedded Systems to be held on October 2, 2006 in Genova, Italy in conjunction with the 9th International Conference on Model Driven Engineering Languages and Systems, MoDELS/UML 2006.

W53 Jules White, Andrey Nechypurenko, Egon Wuchner, and Douglas C. Schmidt, "Intelligence Frameworks for Assisting Modelers in Combinatorically Challenging Domains," Proceedings of the Workshop on Generative Programming and Component Engineering for QoS Provisioning in Distributed Systems, October 23, 2006, Portland, Oregon.

W52 Nishanth Shankaran, Xenofon Koutsoukos, Douglas C. Schmidt, and Aniruddha Gokhale, "Evaluating Adaptive Resource Management for Distributed Real-Time Embedded Systems," Proceedings of the 4th Workshop on Adaptive and Reflective Middleware, November 28, 2005 Grenoble, France.

W51 Jules White and Douglas Schmidt, "Simplifying the Development of Product-line Customization Tools via Model Driven Development," MODELS 2005 workshop on MDD for Software Product-lines: Fact or Fiction? , October 2, 2005, Jamaica.

W50 Arvind S. Krishna, Aniruddha Gokhale, Douglas C. Schmidt, Venkatesh Prasad Ranganath, and John Hatcliff, "Model-driven Middleware Specialization Techniques for Software Product-line Architectures in Distributed Real-time and Embedded Systems," MODELS 2005 workshop on MDD for Software Product-lines: Fact or Fiction? , October 2, 2005, Jamaica.

W49 Gen Deng, Gunther Lenz, and Douglas C. Schmidt, "Addressing Domain Evolution Challenges in Model-Driven Software Product-line Architectures," MODELS 2005 workshop on MDD for Software Product-lines: Fact or Fiction? , October 2, 2005, Jamaica.

W48 Andrey Nechypurenko and Douglas C. Schmidt, "Supporting Model Reusability with Pattern-based Composition Units," Proceedings of the IST 2nd European Workshop on Model Driven Architecture (MDA), with an emphasis on Methodologies and Transformations September 7th-8th 2004, Canterbury, England.

W47 Cemal Yilmaz, Arvind S. Krishna, Atif Memon, Adam Porter, Douglas C. Schmidt, Aniruddha Gokhale, and Balachandran Natarajan, "A Model-based Distributed Continuous Quality Assurance Process to Enhance the Quality of Service of Evolving Performance-intensive Software Systems," Proceedings of the 2nd ICSE Workshop on Remote Analysis and Measurement of Software Systems (RAMSS), Edinburgh, Scotland, UK, May 24, 2004.

W46 Andrey Nechypurenko, Douglas C. Schmidt, Tao Lu, Gan Deng, Aniruddha Gokhale, "Applying MDA and Component Middleware to Large-scale Distributed Systems: a Case Study, Proceedings of the OMG 1st European Workshop on Model Driven Architecture with Emphasis on Industrial Application, Enschede, the Netherlands, March 2004.

W45 Gan Deng, Tao Lu, Emre Turkay, Aniruddha Gokhale, Douglas C. Schmidt, and Andrey Nechypurenko, "Model Driven Development of Inventory

Tracking System," Proceedings of the OOPSLA 2003 Workshop on Domain-Specific Modeling Languages, Anaheim, CA, October 2003.

W44 Tao Lu, Emre Turkay, Aniruddha Gokhale, and Douglas C. Schmidt, "CoSMIC: An MDA Tool suite for Application Deployment and Configuration," Proceedings of the OOPSLA 2003 Workshop on Generative Techniques in the Context of Model Driven Architecture, Anaheim, CA, October 2003.

W43 Arvind S. Krishna, Jai Balasubramanian, Aniruddha Gokhale, Douglas C. Schmidt, Diego Sevilla, and Gautham Thaker, "Empirically Evaluating CORBA Component Model Implementations," Proceedings of the ACM OOPSLA 2003 Workshop on Middleware Benchmarking, Anaheim, CA, October 2003.

W42 Aniruddha Gokhale, Douglas C. Schmidt, Tao Lu, Balachandran Natarjan, and Nanbor Wang, CoSMIC: An MDA Generative Tool for Distributed Real-time and Embedded Applications, Workshop on Model-driven Approaches to Middleware Applications Development at 4th IFIP/ACM/USENIX International Conference on Middleware for Distributed Systems Platforms, June 16, 2003, Rio de Janeiro, Brazil.

W41 Ossama Othman, Jaigaesh Balasubramanian, and Douglas C. Schmidt, "The Design and Performance of an Adaptive Middleware Load Balancing and Monitoring Service," Third International Workshop on Self-Adaptive Software, Arlington, VA, USA, June 9-11, 2003.

W40 Radu Cornea, Shivajit Mohapatra, Nikil Dutt, Rajesh Gupta, Ingolf Krueger, Alex Nicolau, Doug Schmidt, Sandeep Shukla, and Nalini Venkatasubramanian, "A Model-Based Approach to System Specification for Distributed Real-time and Embedded Systems," IEEE RTAS Workshop on Model-Driven Embedded Systems, Washington DC, May 27-30, 2003.

W39 Adam Porter, Cemal Yilmaz, and Douglas C. Schmidt "Distributed Continuous Quality Assurance: The Skoll Project," Proceedings of the Workshop on Remote Analysis and Measurement of Software Systems (RAMSS), Portland, Oregon, May 9, 2003.

W38 Krishnakumar Balasubramanian, Douglas C. Schmidt, Nanbor Wang, Christopher D. Gill, "Towards Composable Distributed Real-time and Embedded Software," Proceedings of the 8th IEEE Workshop on Object-oriented Real-time Dependable Systems, Guadalajara, Mexico, January 2003.

W37 Aniruddha Gokhale, Balachandran Natarjan, Douglas C. Schmidt, Andrey Nechypurenko, Nanbor Wang, Jeff Gray, Sandeep Neema, Ted Bapty, and Jeff Parsons, "CoSMIC: An MDA Generative Tool for Distributed Real-time and Embedded Component Middleware and Applications," Proceedings of the OOPSLA 2002 Workshop on Generative Techniques in the Context of Model Driven Architecture, Seattle, WA, November 2002.

W36 M. Mousavi, G. Russello, M. Chaudron, M. Reniers, T. Basten, A. Corsaro, S. Shukla, R. Gupta, and D.C. Schmidt, "Using Aspect-GAMMA in Design and Verification of Embedded Systems," Proceedings of the Seventh Annual IEEE

International Workshop on High Level Design Validation and Test Workshop, Cannes, France, October 27–29, 2002.

W35 Douglas C. Schmidt, Andy Gokhale, and Chris Gill, "Applying Model-Integrated Computing and DRE Middleware to High Performance Embedded Computing Applications," Proceedings of the 6th Annual Workshop on High-Performance Embedded Computing (HPEC), September 24–26, Boston, MA.

W34 Douglas C. Schmidt, "Adaptive and Reflective Middleware for Distributed Real-time and Embedded Systems," EMSOFT 2002: Second Workshop on Embedded Software, Grenoble, France, October, 7–9th, 2002.

W33 Aniruddha S. Gokhale and Douglas C. Schmidt and Joseph K. Cross and Christopher Andrews and Sylvester J. Fernandez and Bala Natarajan and Nanbor Wang and Chris D. Gill, "Towards Real-time Support in Fault-tolerant CORBA," IEEE Workshop on Dependable Middleware-Based Systems, Washington, D.C., June 23-26, 2002.

W32 Chris Gill, Joe Loyall, Rick Schantz, and Douglas C. Schmidt, "Lessons Learned From Using Adaptive DOC Middleware in Real Application Contexts," IEEE Workshop on Dependable Middleware-Based Systems, Washington, D.C., June 23-26, 2002.

W31 M. Mousavi, G. Russello, M. Chaudron, M. Reniers, T. Basten, A. Corsaro, S. Shukla, R. Gupta, and D.C. Schmidt, "Aspects + GAMMA = AspectGAMMA A Formal Framework for Aspect-Oriented Specification," Proceedings of Early Aspects: Aspect-Oriented Requirements Engineering and Architecture Design Workshop, Enschede, The Netherlands, April 2002.

W30 Joseph K. Cross and Douglas C. Schmidt, "Meta-Programming Techniques for Distributed Real-time and Embedded Systems," Proceedings of the 7th IEEE Workshop on Object-oriented Real-time Dependable Systems, San Diego, CA, January, 2002.

W29 Douglas C. Schmidt and Mayur Deshpande and Carlos O'Ryan, "Operating System Performance in Support of Real-time Middleware," Proceedings of the 7th IEEE Workshop on Object-oriented Real-time Dependable Systems, San Diego, CA, January, 2002.

W28 Christopher D. Gill, Ron Cytron, and Douglas C. Schmidt, "Middleware Scheduling Optimization Techniques for Distributed Real-Time and Embedded Systems," Proceedings of the 7th IEEE Workshop on Object-oriented Real-time Dependable Systems, San Diego, CA, January, 2002.

W27 Douglas C. Schmidt, "Adaptive and Reflective Middleware for Distributed Real-time and Embedded Systems," EMSOFT 2001: First Workshop on Embedded Software, Lake Tahoe, California, October, 8th–10th, 2001.

W26 Darrell Brunsch, Carlos O'Ryan, and Douglas C. Schmidt, "Designing an Efficient and Scalable Server-side Asynchrony Model for CORBA," Proceedings of the ACM SIGPLAN Workshop on Optimization of Middleware and Distributed Systems (OM 2001), Snowbird, Utah, June 18, 2001.

W25 Irfan Pyarali, Marina Spivak, Douglas C. Schmidt, and Ron Cytron, "Optimizing Thread-Pool Strategies for Real-Time CORBA," Proceedings of

the ACM SIGPLAN Workshop on Optimization of Middleware and Distributed Systems (OM 2001), Snowbird, Utah, June 18, 2001.

W24 Yamuna Krishnamurthy, Vishal Kachroo, David A. Karr, Craig Rodrigues, Joseph P. Loyall, Richard Schantz, and Douglas C. Schmidt, "Integration of QoS-enabled Distributed Object Computing Middleware for Developing Next-generation Distributed Applications," Proceedings of the ACM SIGPLAN Workshop on Optimization of Middleware and Distributed Systems (OM 2001), Snowbird, Utah, June 18, 2001.

W23 Ossama Othman and Douglas C. Schmidt, Optimizing Distributed system Performance via Adaptive Middleware Load Balancing, Proceedings of the ACM SIGPLAN Workshop on Optimization of Middleware and Distributed Systems (OM 2001), Snowbird, Utah, June 18, 2001.

W22 Pradeep Gore, Douglas C. Schmidt, Carlos O'Ryan, and Ron Cytron, "Designing and Optimizing a Scalable CORBA Notification Service," Proceedings of the ACM SIGPLAN Workshop on Optimization of Middleware and Distributed Systems (OM 2001), Snowbird, Utah, June 18, 2001.

W21 Douglas C. Schmidt and Adam Porter, "Leveraging Open-Source Processes to Improve the Quality and Performance of Open-Source Software," Proceedings of the 1st Workshop on Open Source Software Engineering, ICSE 23, Toronto, Canada, May 15, 2001.

W20 Christopher D. Gill, David Levine, Douglas C. Schmidt, "Towards Real-Time Adaptive QoS Management in Middleware for Embedded Computing Systems," Fourth Annual Workshop on High Performance Embedded Computing, MIT Lincoln Laboratory, September 20-22, 2000.

W19 Christopher D. Gill, Fred Kuhns, David Levine, Douglas C. Schmidt, Bryan S. Doerr, and Richard E. Schantz, "Applying Adaptive Real-time Middleware to Address Grand Challenges of COTS-based Mission-Critical Real-Time Systems," Proceedings of the 1st International Workshop on Real-Time Mission-Critical Systems: Grand Challenge Problems, IEEE, Phoenix, Arizona, November 30, 1999.

W18 Carlos O'Ryan, Douglas C. Schmidt, David Levine, and Russell Noseworthy, "Applying a Real-time CORBA Event Service to Large-scale Distributed Interactive Simulation," 5th International Workshop on Object-oriented Real-Time Dependable Systems, IEEE, Monterey, CA, November 15-18, 1999.

W17 Fred Kuhns, Carlos O'Ryan, Douglas C. Schmidt, and Jeff Parsons, "The Design and Performance of a Pluggable Protocols Framework for Object Request Broker Middleware," Proceedings of the IFIP Sixth International Workshop on Protocols For High-Speed Networks (PfHSN '99), Salem, MA, August 25–27, 1999.

W16 David Levine, Sergio Flores-Gaitan, and Douglas C. Schmidt, "Measuring OS Support for Real-time CORBA ORBs," Proceedings of the Fourth International IEEE Workshop on Object-oriented Real-time Dependable Systems (WORDS'99), Santa Barbara, California, January 27-29, 1999.

W15 Douglas C. Schmidt, Rajeev Bector, David Levine Sumedh Mungee, and Guru Parulkar, "TAO: a Middleware Framework for Real-time ORB Endsystems," Proceedings of the Workshop on Middleware for Real-Time Systems and Services, held in conjunction with IEEE Real-time Systems Symposium, San Francisco, CA, December 2nd, 1997.

W14 Aniruddha Gokhale, Tim Harrison, Douglas C. Schmidt, and Guru Parulkar, "Operating System Support for Real-time CORBA," *Proceedings of the 5th International Workshop on Object-Orientation in Operating Systems*: IWOOOS 1996 workshop, October 27–28, 1996, Seattle, Washington.

W13 Douglas C. Schmidt, Guru Parulkar, and Chuck Cranor, "Gigabit CORBA - High-Performance Distributed Object Computing," Gigabit Networking Workshop (GBN'96), 24 March 1996, San Francisco, in conjunction with INFOCOM '96.

W12 Douglas C. Schmidt, "Acceptor and Connector: Design Patterns for Actively and Passively Initializing Network Services." Workshop on Pattern Languages of Object-Oriented Programs at ECOOP '95, August 7 – 1, 1995, Aarhus, Denmark.

W11 Douglas C. Schmidt, "High-Performance Event Filtering for Dynamic Multi-point Applications," Proceedings of the 1st Workshop on High Performance Protocol Architectures (HIPPARCH), INRIA, Sophia Antipolis, France, December, 1994, p 1–8.

W10 Douglas C. Schmidt, "Flexible Configuration of High-Performance Object-Oriented Distributed Communication Systems," 9th OOPSLA Conference, invited paper to the Workshop on Flexibility in Systems Software, ACM, Portland, Oregon, October, 1994, pp. 1–4.

W9 Douglas C. Schmidt and Tatsuya Suda, "Measuring the Impact of Alternative Parallel Process Architectures on Communication Subsystem Performance," *Proceedings of the Proceedings of the 4th International Workshop on Protocols for High-Speed Networks*, IFIP, Vancouver, British Columbia, August, 1994, pp. 103–118.

W8 Douglas C. Schmidt and Tatsuya Suda, "The Service Configurator Framework: An Extensible Architecture for Dynamically Configuring Concurrent, Multi-service Network Daemons," *Proceedings of the 2nd International Workshop on Configurable Distributed Systems*, IEEE, Pittsburgh, PA, March 21-23, 1994, pp. 190–201.

W7 Douglas C. Schmidt, Burkhard Stiller, Tatsuya Suda, and Martina Zitterbart, "Tools for Generating Application-Tailored Multimedia Protocols on Heterogeneous Multi-Processor Platforms," *Proceedings of the 2nd Workshop on High-Performance Communications Subsystems*, IEEE, Williamsburg, Virginia, September 1-3, 1993, pp. 1–7.

W6 Douglas C. Schmidt and Tatsuya Suda, "A Framework for Developing and Experimenting with Parallel Process Architectures to Support High-Performance Transport Systems," *Proceedings of the 2nd Workshop on*

High-Performance Communications Subsystems, IEEE, Williamsburg, Virginia, September 1-3, 1993, pp. 1–8.

W5 Tatsuya Suda, Douglas C. Schmidt, Donald F. Box, Duke Hong and Hung Huang, "High Speed Networks," *Proceedings of the International Computer World Symposium '92*, Kobe, Japan, November, 1992.

W4 Hung K. Huang, Douglas C. Schmidt, Donald F. Box, Kazu Shimono, Girish Kotmire, Unmesh Rathi, and Tatsuya Suda, "ADAPTIVE: A Prototyping Environment for Transport Systems." *Proceedings of the 4th International Workshop on Computer Aided Modeling, Analysis, and Design of Communication Links and Networks (CAMAD)*, IEEE, Montreal, Canada, September, 1992.

W3 Donald F. Box, Douglas C. Schmidt, and Tatsuya Suda, "Alternative Approaches to ATM/Internet Interoperation," *Proceedings of the 1st Workshop on the Architecture and Implementation of High-Performance Communication Subsystems*, IEEE, Tucson, Arizona, February 17-19, 1992, pp. 1–5.

W2 Douglas C. Schmidt and Richard Selby "Modeling Software Interconnectivity," *Proceedings of the 22nd Symposium on the Interface: Computer Science and Statistics*, East Lansing, MI, May, 1990.

W1 Richard W. Selby, Greg James, Kent Madsen, Joan Mahoney, Adam A. Porter, and Douglas C. Schmidt "Classification Tree Analysis Using the Amadeus Measurement and Empirical Analysis System," *Proceedings of the 14th Annual Software Engineering Workshop at NASA Software Engineering Laboratory*, College Park, Maryland, November, 1989, pp. 239–250.

- **Trade Magazine and Newsletter Publications**

M73 Douglas C. Schmidt, Accelerating the Industrial Internet with the OMG Data Distribution Service, Real-time Innovations, January 2014.

M72 Douglas C. Schmidt Ron Guida, "Elastic Application Platforms for Cloud Computing," HPC In the Cloud, September 2010.

M71 Douglas C. Schmidt Ron Guida, "Achieving Ultra High Performance in the Cloud," HPC In the Cloud, August 2010.

M70 Douglas C. Schmidt, "Building Ultra High Performance Computing Applications with Zircon Software," Programmer's Paradise, May 31st, 2010.

M69 Egon Wuchner, Andrey Nechypurenko, Jules White, and Douglas C. Schmidt, "Das Generic Eclipse Modeling System (GEMS): Skalierbare Domanenmodellierung Leicht(er) Gemacht," SIGS ObjectSpektrum, June, 2007.

M68 Jules White, Douglas C. Schmidt, Andrey Nechypurenko, and Egon Wuchner, Introduction to the Generic Eclipse Modeling System, Eclipse Magazine, Volume 06, January, 2007.

M67 Bala Natarajan, Douglas C. Schmidt, and Steve Vinoski, "Object Interconnections: The CORBA Component Model Part 4: Implementing Components with CCM, C/C++ Users Journal, October, 2004.

M66 Bala Natarajan, Douglas C. Schmidt, and Steve Vinoski, "The CORBA Component Model Part 3: The CCM Container Architecture and Component Implementation Framework," C/C++ Users Journal, September, 2004.
M65 Douglas C. Schmidt, Richard Schantz, Aniruddha Gokhale, and Joe Loyall, "Middleware R&D Challenges for Distributed Real-time and Embedded Systems," SIGBED Review, Volume 1, No. 1, April 2004.
M64 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: The CORBA Component Model: Part 2: Defining Components with the IDL 3.x Types," C/C++ Users Journal, May, 2004.
M63 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: The CORBA Component Model: Part 1: Evolving Towards Components," C/C++ Users Journal, February, 2004.
M62 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: XML Reflection for CORBA," C/C++ Users Journal, December, 2003.
M61 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: CORBA Metaprogramming Mechanisms, Part 1: Portable Interceptors Concepts and Components," C/C++ Users Journal, March, 2003.
M60 Douglas C. Schmidt and Steve Vinoski, Object Interconnections: "Dynamic CORBA, Part 4: The Interface Repository," C/C++ Users Journal, January, 2003.
M59 Douglas C. Schmidt, "Voice of the Customer: An Interview," Raytheon Technology Today, spring 2003, Volume 2, Issue 1.
M58 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: Dynamic CORBA, Part 3: The Dynamic Skeleton Interface," C/C++ Users Journal, November, 2002.
M57 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: Dynamic CORBA, Part 2: Dynamic Any," C/C++ Users Journal, September, 2002.
M56 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: Dynamic CORBA, Part 1: The Dynamic Invocation Interface," C/C++ Users Journal, July, 2002.
M55 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: Real-time CORBA, Part 4: Protocol Selection and Explicit Binding," C/C++ Users Journal, May, 2002.
M54 Douglas C. Schmidt and Steve Huston, "Why Standards Alone Won't Get You Portable Software And How to Make Open Source Development Work for You," InformIT: Focus on C++, Addison-Wesley.
M53 Douglas C. Schmidt and Steve Vinoski, "Real-time CORBA Part 3: Thread Pools and Synchronizers," C/C++ Users Journal, March, 2002.
M52 Douglas C. Schmidt and Steve Vinoski, "Real-time CORBA, Part 2: Applications and Priorities," C/C++ Users Journal, January, 2002.
M51 Douglas C. Schmidt and Steve Vinoski, "Real-time CORBA, Part 1: Motivation and Overview," *C/C++ Users Journal*, October, 2001.
M50 Douglas C. Schmidt and Steve Vinoski, "CORBA and XML, Part 3: SOAP and Web Services," *C/C++ Users Journal*, September, 2001.

M48 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: CORBA and XML, Part 2: XML as CORBA Data," *C/C++ Users Journal*, July, 2001.
M47 Douglas C. Schmidt and Steve Vinoski, "Object Interconnections: CORBA and XML, Part 1: Versioning," *C/C++ Users Journal*, May, 2001.
M46 Douglas C. Schmidt and Steve Vinoski, "Standard C++ and the OMG C++ Mapping: Server-side Mappings and Pseudo-Objects," *C/C++ Users Journal*, April, 2001.
M45 Douglas C. Schmidt and Steve Vinoski, "Standard C++ and the OMG C++ Mapping," *C/C++ Users Journal*, January, 2001.
M44 Douglas C. Schmidt and Steve Vinoski, "The History of the OMG C++ Mapping", *C/C++ Users Journal*, November, 2000.
M43 Douglas C. Schmidt and Steve Vinoski, "An Overview of the OMG CORBA Messaging Quality of Service (QoS) Framework," *C++ Report*, SIGS, Vol. 12, No 3, March, 2000.
M42 Douglas C. Schmidt, "Monitor Object – an Object Behavior Pattern for Concurrent Programming," *C++ Report*, SIGS, Vol. 12., No. 4. May 2000.
M41 Alexander B. Arulanthu, Carlos O'Ryan, Douglas C. Schmidt, and Michael Kircher, "Applying Patterns and Components to Develop an IDL Compiler for CORBA AMI Callbacks," *C++ Report*, SIGS, Vol. 12, No. 3, March, 2000.
M40 David Levine, Chris Gill, and Douglas C. Schmidt, "Object Lifetime Manager – A Complementary Pattern for Controlling Object Creation and Destruction," *C++ Report*, SIGS, Vol. 11, No. 11, November/December, 1999.
M39 Douglas C. Schmidt, Steve Vinoski, and Nanbor Wang, "Collocation Optimizations for CORBA," *C++ Report*, SIGS, Vol. 11, No. 10, October, 1999.
M38 Douglas C. Schmidt, "Strategized Locking, Thread-safe Decorator, and Scoped Locking: Patterns and Idioms for Simplifying Multi-threaded C++ Components,"*C++ Report*, SIGS, Vol. 11, No. 9, September, 1999.
M37 Douglas C. Schmidt and Steve Vinoski, "Time-Independent Invocation and Interoperable Routing," *C++ Report*, SIGS, Vol. 11, No 5, May, 1999.
M36 Michael Kircher and Douglas C. Schmidt, "Dove: A Distributed Object Visualization Environment," *C++ Report*, SIGS, Vol. 11, No 3, March, 1999.
M35 Douglas C. Schmidt, "Wrapper Facade: A Structural Pattern for Encapsulating Functions within Classes," *C++ Report*, SIGS, Vol. 11, No 2, February, 1999.
M34 Douglas C. Schmidt and Steve Vinoski, "Programming Asynchronous Method Invocation with CORBA Messaging," *C++ Report*, SIGS, Vol. 11, No 2, February, 1999.
M33 Douglas C. Schmidt "Why Software Reuse has Failed and How to Make It Work for You," C++ Report, SIGS, Vol. 11, No. 1, January, 1999.
M32 Douglas C. Schmidt, "An Architectural Overview of the ACE Framework: A Case-study of Successful Cross-platform Systems Software Reuse," USENIX login magazine, Tools special issue, November, 1998.

M31 Douglas C. Schmidt, "GPERF: A Perfect Hash Function Generator," C++ Report, SIGS, Vol. 10, No. 10, November/December, 1998.
M30 Douglas C. Schmidt and Steve Vinoski, "Introduction to CORBA Messaging," SIGS, Vol. 10, No 10, November/December, 1998.
M29 Douglas C. Schmidt and Steve Vinoski, "C++ Servant Managers for the Portable Object Adapter," SIGS, Vol. 10, No 8, September, 1998.
M28 Chris Cleeland and Douglas C. Schmidt, "External Polymorphism, An Object Structural Pattern for Transparently Extending C++ Concrete Data Types," C++ Report, SIGS, Vol. 10, No. 6, July/August, 1998.
M27 Douglas C. Schmidt and Irfan Pyarali, "Strategies for Implementing POSIX Condition Variables on Win32," C++ Report, SIGS, Vol. 10, No. 5, June, 1998.
M26 Douglas C. Schmidt and Steve Vinoski, "Using the Portable Object Adapter for Transient and Persistent CORBA Objects," C++ Report, SIGS, Vol. 10, No 4. May, 1998.
M25 Douglas C. Schmidt, "Applying Design Patterns to Simplify Signal Handling," C++ Report, SIGS, Vol. 10, No. 4, May, 1998.
M24 Douglas C. Schmidt, Tim H. Harrison, and Nat Pryce, "Thread-specific Storage: an Object Behavioral Pattern for Efficiently Accessing per-Thread State," C++ Report, SIGS, Vol. 9, No. 10, November/December, 1997
M23 Douglas C. Schmidt and Steve Vinoski, "Object Adapters: Concepts and Terminology," C++ Report, SIGS, Vol. 9, No 11. November/December, 1997.
M22 Prashant Jain and Douglas C. Schmidt, "Dynamically Configuring Communication Services with the Service Configurator Pattern," *C++ Report*, SIGS, Vol. 9, No. 6, June, 1997.
M21 Douglas C. Schmidt and Steve Vinoski, "Overcoming Drawbacks in the OMG Events Service," *C++ Report*, SIGS, Vol. 9, No 6. June, 1997.
M20 Douglas C. Schmidt and Steve Vinoski, "OMG Event Object Service," *C++ Report*, SIGS, Vol. 9, No 2. February, 1997.
M19 Prashant Jain and Douglas C. Schmidt, "Experiences Converting a C++ Communication Framework to Java," *C++ Report*, SIGS, Vol. 9, No. 1, January, 1996.
M18 Douglas C. Schmidt, "Lessons Learned Building Reusable OO Telecommunication Software," Multiuse Express, Lucent Technologies, Vol. 4, No. 6, December, 1996.
M17 Douglas C. Schmidt and Steve Vinoski, "Distributed Callbacks and Decoupled Communication in CORBA," *C++ Report*, SIGS, Vol. 8, No 9. October, 1996.
M16 Timothy H. Harrison and Douglas C. Schmidt, "Evaluating the Performance of OO Network Programming Toolkits," *C++ Report*, SIGS, Vol. 8, No 7. July/August 1996.
M15 Douglas C. Schmidt and Steve Vinoski, "Comparing Alternative Programming Techniques for Multi-threaded Servers – the Thread-per-Session Concurrency Model," *C++ Report*, SIGS, Vol. 8, No 7. July/August 1996.

M14 Douglas C. Schmidt and Steve Vinoski, "Comparing Alternative Programming Techniques for Multi-threaded Servers – the Thread-Pool Concurrency Model," *C++ Report*, SIGS, Vol. 8, No 4. April 1996.
M13 Douglas C. Schmidt and Steve Vinoski, "Comparing Alternative Programming Techniques for Multi-threaded Servers – the Thread-per-Request Concurrency Model," *C++ Report*, SIGS, Vol. 8, No 2. February 1996.
M12 Douglas C. Schmidt, "A Design Pattern for Actively Initializing Network Services," *C++ Report*, SIGS, Vol. 8, No. 1, January 1996.
M11 Douglas C. Schmidt, "Design Patterns for Initializing Network Services: Introducing the Acceptor and Connector Patterns," *C++ Report*, SIGS, Vol. 7, No. 9, November/December 1995.
M10 Douglas C. Schmidt and Steve Vinoski, "Comparing Alternative Server-side Distributed Programming Techniques," Object Interconnections Column, *C++ Report*, SIGS, Vol. 7, No. 8, October 1995.
M9 Douglas C. Schmidt and Steve Vinoski, "Comparing Alternative Client-side Distributed Programming Techniques," Object Interconnections Column, *C++ Report*, SIGS, Vol. 7, No. 4, May 1995.
M8 Douglas C. Schmidt and Paul Stephenson, "Using Design Patterns to Evolve System Software from UNIX to Windows NT," *C++ Report*, SIGS, Vol. 7, No. 3, March/April 1995, pp. 27–39.
M8 Douglas C. Schmidt and Steve Vinoski, "Distributed Object Computing by Example," Object Interconnections Column, *C++ Report*, SIGS, Vol. 7, No. 2, February 1995.
M7 Douglas C. Schmidt and Steve Vinoski, "Distributed Object Computing with C++," Object Interconnections Column, *C++ Report*, SIGS, Vol. 7, No. 1, January 1995.
M6 Douglas C. Schmidt, "Transparently Parameterizing Synchronization Mechanisms into a Concurrent Distributed Application," *C++ Report*, SIGS, Vol. 6, No. 5, July/August 1994, pp. 1–10.
M5 Douglas C. Schmidt, "A Domain Analysis of Network Daemon Design Dimensions," *C++ Report*, SIGS, Vol. 6, No. 3, March/April, 1994, pp. 1–12.
M4 Douglas C. Schmidt, "The Object-Oriented Design and Implementation of the Reactor: A C++ Wrapper for UNIX I/O Multiplexing," *C++ Report*, SIGS, Vol. 5, No. 7, September, 1993, pp. 1–14.
M3 Douglas C. Schmidt, "The Reactor: An Object-Oriented Interface for Event-Driven UNIX I/O Multiplexing," *C++ Report*, SIGS, Vol. 5, No. 2, February, 1993, pp. 1–12.
M2 Douglas C. Schmidt, "IPC_SAP: An Object-Oriented Interface to Operating System Interprocess Communication Services," *C++ Report*, SIGS, Vol. 4, No. 8, November/December, 1992, pp. 1–10.
M1 Douglas C. Schmidt, "Systems Programming with C++ Wrappers: Encapsulating Interprocess Communication Services with Object-Oriented Interfaces," *C++ Report*, SIGS, Vol. 4, No. 7, September/October, 1992, pp 1–6.

- **Refereed Short Papers, Posters, and Demos**

P13 Akram Hakiri, Berthou Pascal, Gayraud Thierry, Aniruddha Gokhale, Joe Hoffert, and Douglas C. Schmidt, "SIP-based QoS Support and Session Management for DDS-based Distributed Real-time and Embedded Systems," Poster Proceedings of the 5th ACM International Conference on Distributed Event-based Systems (DEBS'11), New York City, NY, USA, July 11-15, 2011.

P12 Jules White, Andrey Nechypurenko, Egon Wuchner, and Douglas C. Schmidt "Automatic Role-based Constraint Solving for Real-Time and Embedded Systems: An Approach to Modeling Guidance", poster paper at the 14th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS), March 26th-29th, 2007, Tucson, Arizona.

P11 John Kinnebrew, Nishanth Shankaran, Gautam Biswas, and Douglas Schmidt, "A Decision-Theoretic Planner with Dynamic Component Reconfiguration for Distributed Real-Time Applications," Poster paper at the Twenty-First National Conference on Artificial Intelligence, July 16 20, 2006, Boston, Massachusetts.

P10 Jai Balasubramanian, Nishanth Shankar, Douglas C. Schmidt, Gautam Biswas, Patrick Lardieri, Ed Mulholland, and Tom Damiano, "A Framework for (Re)Deploying Components in Distributed Realtime and Embedded Systems," poster paper at the Dependable and Adaptive Distributed Systems, Track of the 21st ACM Symposium on Applied Computing, April 23 -27, 2006, Bourgogne University, Dijon, France.

P9 Jules White and Douglas C. Schmidt, Reducing Enterprise Product Line Architecture Deployment Costs via Model-Driven Deployment and Configuration Testing, Poster paper at the 13th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS '06) March 27th-30th, 2006, University of Potsdam, Potsdam, Germany.

P8 Arvind S. Krishna, Aniruddha Gokhale, Douglas C. Schmidt, John Hatcliff, and Venkatesh Prasad Ranganat, "Towards Highly Optimized Real-time Middleware for Software Product-line Architectures," Proceedings of the Work-In-Progress session at the 26th IEEE Real-Time Systems Symposium, December 5-8, 2005, Miami, Florida.

P7 Gan Deng, Douglas Schmidt, and Aniruddha Gokhale, "Supporting Configuration and Deployment of Component-based DRE Systems Using Frameworks, Models, and Aspects," Poster Session of the 20th Annual ACM SIGPLAN Conference on Object-Oriented Programing, Systems, Languages and Applications (OOPSLA 2005), San Diego, CA, October 16-20, 2005.

P6 Jules White, Douglas Schmidt, and Aniruddha Gokhale, "The J3 Process for Building Autonomic Enterprise Java Bean Systems," Proceedings of the International Conference on Autonomic Computing (ICAC 2005), Seattle, WA, June 2005 (short paper).

P5 Aniruddha Gokhale, Arvind Krishna and Douglas C. Schmidt, "CoSMIC: Addressing Crosscutting Deployment and Configuration Concerns of Distributed Real-time and Embedded Systems via Aspect-oriented and Model-driven Software Development," Demo Session, Fourth International Conference on Aspect-oriented Software Development (AOSD), Chicago, IL, Mar 2005.

P4 Aniruddha Gokhale, Krishnakumar Balasubramanian, Jaiganesh Balasubramanian, Arvind Krishna, George Edwards, Gan Deng, Jeff Parsons, Tao Lu, Balachandran Natarajan and Douglas C. Schmidt, "CoSMIC: Addressing Crosscutting Deployment and Configuration Concerns in QoS-sensitive Distributed Systems," 19th ACM OOPSLA Conference, Poster Session, Vancouver, Canada, Oct 2004.

P3 Joseph K. Cross and Douglas C. Schmidt, "Meta-Programming Techniques to Declaratively Optimize Middleware Policies and Mechanisms," Poster session at the IFIP/ACM Middleware 2001 International Conference on Distributed Systems Platforms, Heidelberg, Germany, November 12-16, 2001.

P2 Aniruddha Gokhale and Douglas C. Schmidt, "Design Principles and Optimizations for High Performance ORBs," ACM, *OOPSLA 97*, Poster Session, Oct 1997, Atlanta, GA, USA.

P1 Douglas C. Schmidt, "Performance Experiments on Alternative Methods for Structuring Active Objects in High-Performance Parallel Communication Systems," 9th ACM OOPSLA Conference, poster paper, Portland, Oregon, October, 1994, pp. 1–12.

- **Editorials and Book Forewords**

E63 Douglas C. Schmidt, "The Top 10 Blog Posts of 2015: Technical Debt, DevOps, Graph Analytics, Secure Coding, and Testing," January 4th, 2016.

E62 Carol Sledge and Douglas C. Schmidt, "A Discussion on Open-Systems Architecture," SEI Blog November 23rd, 2015.

E61 Douglas C. Schmidt, "Agile, Architecture Fault Analysis, the BIS Wassenaar Rule, and Computer Network Design," SEI Blog, September 7, 2015.

E60 Douglas C. Schmidt, "Testing, Agile Metrics, Fuzzy Hashing, Android, and Big Data" SEI Blog, July 13, 2015.

E59 Doug,as C. Schmidt, "Resilience, Model-Driven Engineering, Software Quality, and Android App Analysis," SEI Blog, May 18, 2015.

E58 Douglas C. Schmidt, "Resilience, Metrics, Sustainment, and Software Assurance," SEI Blog, February 23, 2015.

E57 Douglas C. Schmidt, "Software Assurance, Social Networking Tools, Insider Threat, and Risk Analysis," SEI Blog, January 19th 2015.

E56 Douglas C. Schmidt, "The 2014 Year in Review," SEI blog, December 22nd, 2014.

E55 Douglas C. Schmidt, Android, Heartbleed, Testing, and DevOps: An SEI Blog Mid-Year Review, SEI blog, June 30th, 2014.

E54 Douglas C. Schmidt, "The Importance of Automated Testing in Open Systems Architecture Initiatives," SEI blog, March 23rd, 2014.
E53 Douglas C. Schmidt, "How Vanderbilt's Secret Software Lab Is Saving America," gizmodo.com, January 10th, 2014.
E52 Douglas C. Schmidt, "The SEI Blog: the Research Year in Review," SEI blog, December 23rd, 2013.
E51 Douglas C. Schmidt "The Architectural Evolution of DoD Combat Systems," SEI blog, November 25th, 2013.
E50 Douglas C. Schmidt, "Three Qs: Vanderbilt Professor Douglas Schmidt," GE's Industrial Internet blog, September 10th, 2013.
E49 Douglas C. Schmidt, "Towards Affordable DoD Combat Systems in the Age of Sequestration," SEI Blog, September 9th, 2013.
E48 Douglas C. Schmidt, "Ten Tech Terms Everyone Needs to Know for 2014," Yahoo Tech News, August 12, 2013.
E47 Douglas C. Schmidt and Philippe Fauchet, "Students Must Stay to Better Workforce," The Tennessean, August 6th, 2013.
E46 Douglas C. Schmidt, "Learning in MOOC Years," Vanderbilt Magazine, Spring 2013.
E45 Douglas C. Schmidt, The SEI Blog: A Two-year Retrospective, SEI blog, April 1st, 2013.
E44 Douglas C. Schmidt, 2012: "The Research Year in Review," SEI blog, December 24th, 2012.
E43 Douglas C. Schmidt, "Reflections on 20 Years of Architecture for Distributed Real-time and Embedded Systems by Douglas C. Schmidt," SEI blog, October 29th, 2012.
E42 Douglas C. Schmidt, "Applying Agility to DoD Common Operating Platform Environment Initiatives," SEI blog, July 30th, 2012.
E41 Douglas C. Schmidt, "Balancing Agility and Discipline at Scale," SEI blog, July 23rd, 2012.
E40 Douglas C. Schmidt, "Strategic Management of Architectural Technical Debt," SEI blog, July 16th, 2012.
E39 Douglas C. Schmidt, "Agile Methods: Tools, Techniques, and Practices for the DoD Community," SEI blog, July 9th, 2012.
E38 Douglas C. Schmidt, "Applying Agile at-Scale for Mission-Critical Software-Reliant Systems," SEI blog, July 2nd, 2012.
E37 Douglas C. Schmidt, "Toward Common Operating Platform Environments, Part 2: Understanding Success Drivers," SEI blog, May 7th, 2012.
E36 Douglas C. Schmidt, "Toward Common Operating Platform Environments, Part 1: Doing More for Less," SEI blog, April 30th, 2012.
E35 Douglas C. Schmidt, "The Road Ahead for SEI R&D in 2012," SEI blog, December 26th, 2011.
E34 Douglas C. Schmidt, "A Summary of Key SEI R&D Accomplishments in 2011," SEI blog, December 19th, 2011.

E33 Douglas C. Schmidt, "Bridging the Valley of Disappointment for DoD Software Research with SPRUCE," SEI blog, November 7th, 2011.
E32 Douglas C. Schmidt, "The Growing Importance of Software Sustainment for the DoD, Part 2: SEI R&D Activities Related to Sustaining Software for the DoD", SEI blog, August 15th, 2011.
E31 Douglas C. Schmidt, "The Growing Importance of Software Sustainment for the DoD, Part 1: Current Trends and Challenges", SEI blog, August 1st, 2011.
E30 Douglas C. Schmidt, "New and Upcoming SEI Research Initiatives," SEI blog, February 21st, 2011.
E29 Douglas C. Schmidt, "Advancing the Scope and Impact of SEI Research," SEI blog, February 9th, 2011.
E28 Douglas C. Schmidt, Foreword to the book *Patterns xof Parallel Software Design*, by Jorge Luis Ortega Arjona, Wiley, 2010.
E27 Douglas C. Schmidt, Foreword to the book *Practical Software Factories in .NET* by Gunther Lenz and Christoph Wienands, Apress, 2006.
E26 Douglas C. Schmidt, Guest editorial of the IEEE Computer Special Issue on Model Driven Development, February 2006.
E25 Douglas C. Schmidt, Guest editorial for IEEE Networks magazine Special Issue on Middleware Technologies for Future Communication Networks, January 2004.
E24 Douglas C. Schmidt, Foreword to the book *Fundamentals of Distributed Object Systems: The CORBA Perspective*, by Zahir Tari and Omran Bukhres, Wiley and Sons, 2001.
E23 Douglas C. Schmidt, Foreword to the book *Design Patterns in Communication Software*, edited by Linda Rising and published by Cambridge University Press, 2000.
E22 Douglas C. Schmidt, "Trends in Distributed Object Computing" editorial for the special issue on Distributed Object-Oriented Systems appearing in the Parallel and Distributed Computing Practices journal, edited by Maria Cobb and Kevine Shaw, Vol. 3, No. 1, March 2000.
E21 Douglas C. Schmidt, "Object-Oriented Application Frameworks," guest editorial for the of the Communications of the ACM, Special Issue on Object-Oriented Application Frameworks, Vol. 40, No. 10, October 1997.
E20 Douglas C. Schmidt, "Recent Advances in Distributed Object Computing," guest editorial for the IEEE Communications Magazine feature topic issue on Distributed Object Computing, Vol. 14, No. 2, February, 1997.
E19 Douglas C. Schmidt, Guest editorial for the USENIX Computing Systems Special Issue on Distributed Object Computing Vol. 9, No. 4, November/December, 1996.
E18 Douglas C. Schmidt, "Software Patterns," guest editorial for Communications of the ACM, Special Issue on Patterns and Pattern Languages, Vol. 39, No. 10, October 1996.

E17 Douglas C. Schmidt, "Using Design Patterns to Develop Reuseable Object-Oriented Software," Strategic Directions in Computing Research OO Working Group conference, MIT, June 14-15, 1996.
E16 Douglas C. Schmidt, "The Last Waltz," C++ Report, SIGS, Vol. 11, No. 4, April 1999.
E15 Douglas C. Schmidt, "Patterns++ - the Next Generation," C++ Report, SIGS, Vol. 9, No. 4, April 1997.
E14 Douglas C. Schmidt, "CORBA: CASE for the late '90s? " C++ Report, SIGS, Vol. 9, No. 2, February 1997.
E13 Douglas C. Schmidt, "Java: Friend or Foe," C++ Report, SIGS, Vol. 9, No. 1, January 1997.
E12 Douglas C. Schmidt, "Promise Keepers," C++ Report, SIGS, Vol. 8, No. 11, November/December 1996.
E11 Douglas C. Schmidt, "The Programming Principle," C++ Report, SIGS, Vol. 8, No. 10, October 1996.
E10 Douglas C. Schmidt, "Pattern Forces," C++ Report, SIGS, Vol. 8, No. 9, September 1996.
E9 Douglas C. Schmidt, "The Secrets of Success for C++," C++ Report, SIGS, Vol. 8, No. 9, August 1996.
E8 Douglas C. Schmidt, "The C++ Decade," C++ Report, SIGS, Vol. 8, No. 9, August 1996.
E7 Douglas C. Schmidt, "Addressing the Challenge of Concurrent and Distributed Systems," C++ Report, SIGS, Vol. 8, No. 7, July 1996.
E6 Douglas C. Schmidt, "Delivering the Goods," C++ Report, SIGS, Vol. 8, No. 6, June 1996.
E5 Douglas C. Schmidt, "Problems with Process," C++ Report, SIGS, Vol. 8, No. 5, May 1996.
E4 Douglas C. Schmidt, "The Impact of Social Forces on Software Project Failures," C++ Report, SIGS, Vol. 8, No. 4, April 1996.
E3 Douglas C. Schmidt, "Reality Check," C++ Report, SIGS, Vol. 8, No. 3, March 1996.
E2 Douglas C. Schmidt, "Role Models for Success," C++ Report, SIGS, Vol. 8, No. 2, February 1996.
E1 Douglas C. Schmidt, "A Zest for Programming," C++ Report, SIGS, Vol. 8, No. 1, January 1996.

- **Technical Reports**

TR16 Gan Deng, Douglas C. Schmidt, Aniruddha Gokhale, " Ensuring Deployment Predictability of Distributed Real-time and Embedded Systems," Vanderbilt University, ISIS-07-814, November 2007.
TR15 Jaiganesh Balasubramanian, Sumant Tambe, Chenyang Lu, Christopher Gill, Aniruddha Gokhale, and Douglas C. Schmidt, "FLARe: a Fault-tolerant Lightweight Adaptive Real-time Middleware for Distributed Real-time and Embedded, Systems," Vanderbilt University, ISIS-07-812, October 2007.

TR14 Shanshan Jiang, Yuan Xue, and Douglas Schmidt, "Minimum Disruption Service Composition and Recovery in Mobile Ad hoc Networks," Vanderbilt University, ISIS-06-711, December 2006.

TR13 Andrey Nechypurenko, Egon Wuchner, Jules White, Douglas C. Schmidt, " Application of Aspect-based Modeling and Weaving for Complexity Reduction in the Development of Automotive Distributed Real-time Embedded Systems," Vanderbilt University, ISIS-06-709, July 2006.

TR12 James H. Hill, John M. Slaby, Steve Baker, Douglas C. Schmidt, "Predicting the Behavior for Components of the SLICE Scenario," Vanderbilt University, ISIS-05-608, October 2005.

TR11 Stoyan Paunov, James Hill, Douglas C. Schmidt, John Slaby,and Steve Baker, "Domain-Specific Modeling Languages for Configuring and Evaluating Enterprise DRE System Quality of Service, Vanderbilt University, ISIS-05-606, August 2005.

TR10 John M. Slaby, Steve Baker, James Hill, Doug Schmidt, "Applying System Execution Modeling Tools to Evaluate Enterprise Distributed Real-time and Embedded System QoS," Vanderbilt University, ISIS-05-604, June 2005.

TR9 Fred Kuhns and Carlos O'Ryan and Douglas C. Schmidt and Jeff Parsons, "The Design and Performance of a Pluggable Protocols Framework for Object Request Broker Middleware," Technical Report wucs-99-12, Washington University, St. Louis, MO, Dept. of Computer Science, April 1999.

TR8 Lutz Prechelt, Barbara Unger, Douglas C. Schmidt, "Replication of the First Controlled Experiment on the Usefulness of Design Patterns: Detailed Description and Evaluation." Technical Report wucs-97-34, 77 pgs., Washington University, St. Louis, MO, Dept. of Computer Science, December 1997.

TR7 Aniruddha Gokhale and Douglas C. Schmidt, "Optimizing the Performance of the CORBA Internet Inter-ORB Protocol Over ATM," Washington University technical report #WUCS-97-10.

TR6 James Hu and Sumedh Mungee and Douglas C. Schmidt, "Principles for Developing and Measuring High-performance Web Servers over ATM," Washington University technical report #WUCS-97-10.

TR5 Chris Cleeland, Douglas C. Schmidt, and Tim H. Harrison, "External Polymorphism – An Object Structural Pattern for Transparently Extending Concrete Data Types," The 3rd annual Pattern Languages of Programming conference in Allerton Park, Illinois, September 4-6, 1996, Washington University technical report #WUCS-97-07.

TR4 Timothy H. Harrison, Douglas C. Schmidt, and Irfan Pyarali, "Asynchronous Completion Token," The 3rd annual Pattern Languages of Programming conference in Allerton Park, Illinois, September 4-6, 1996, Washington University technical report #WUCS-97-07.

TR3 Douglas C. Schmidt and Timothy H. Harrison, "The Double-Checked Locking Pattern," The 3rd annual Pattern Languages of Programming conference in

Allerton Park, Illinois, September 4-6, 1996, Washington University technical report #WUCS-97-07.

TR2 Prashant Jain and Douglas C. Schmidt, "The Service Configurator Pattern," The 3rd annual Pattern Languages of Programming conference in Allerton Park, Illinois, September 4-6, 1996, Washington University technical report #WUCS-97-07.

TR1 Douglas C. Schmidt, "Acceptor and Connector: Design Patterns for Initializing Network Services," The EuroPLoP '96 conference in Kloster Irsee, Germany, July 10-14, 1996, Washington University technical report #WUCS-97-07.

Presentations

Conference Presentations

1. "Producing and Delivering a Coursera MOOC on Pattern-Oriented Software Architecture for Concurrent and Networked Software," WaveFront forum at the SPLASH 2013 conference, Indianapolis, IN, October 29th, 2013.
2. "Addressing the Challenges of Tactical Information Management in Net-Centric Systems with the OMG Data Distribution Service (DDS)," the 16th International ACM Workshop on Parallel and Distributed Real-Time Systems (WPDRTS '08), Miami, Florida, April 14, 2008.
3. "The Design and Performance of Configurable Component Middleware for End-to-End Adaptation of Distributed Real-time Embedded Systems," proceedings of the 10th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing (ISORC), May 7-9, 2007, Santorini Island, Greece.
4. "A Decision-Theoretic Planner for DRE Systems," 7th OMG Real-time/Embedded CORBA workshop, Arlington, VA, July 10–13, 2006.
5. "Model-driven QoS Provisioning for Real-time CORBA and CCM DRE Systems," 6th OMG Real-time/Embedded CORBA workshop, Arlington, VA, July 11–14, 2005.
6. "Research Advances in Middleware for Distributed Systems: State of the Art," Computer Communications stream of the 17th IFIP World Computer Congress, Montreal, Canada, August 25-30, 2002.
7. "Towards Highly Configurable Real-time Object Request Brokers," the IEEE International Symposium on Object-Oriented Real-time Distributed Computing (ISORC), Washington DC, April 29 – May 1, 2002.
8. "Operating System Performance in Support of Real-time Middleware," Proceedings of the 7th IEEE Workshop on Object-oriented Real-time Dependable Systems, San Diego, CA, January, 2002.
9. "Designing an Efficient and Scalable Server-side Asynchrony Model for CORBA," Proceedings of the ACM SIGPLAN Workshop on Optimization of Middleware and Distributed Systems (OM 2001), Snowbird, Utah, June 18, 2001.

10. "DOORS: Towards High-performance Fault-Tolerant CORBA," Proceedings of the 2nd International Symposium on Distributed Objects and Applications (DOA '00), OMG, Antwerp, Belgium, September 2000.
11. "The Design and Performance of a CORBA Audio/Video Streaming Service," Proceedings of the 31st Hawaii International Conference on System Systems (HICSS), Hawaii, January, 1999, minitrack on Multimedia DBMS and the WWW, Hawaii, January 1999.
12. "Alleviating Priority Inversion and Non-determinism in Real-time CORBA ORB Core Architectures," Proceedings of the Fourth IEEE Real-Time Technology and Applications Symposium (RTAS), Denver, Colorado, June 3-5, 1998
13. "Optimizing the Performance of the CORBA Internet Inter-ORB Protocol Over ATM," Proceedings of the 31st Hawaii International Conference on System Systems (HICSS), Hawaii, January, 1998. This was selected as the best paper in the Software Technology Track (188 submitted, 77 accepted).
14. "The Double-Checked Locking Pattern," *Proceedings of the 3rd annual Pattern Languages of Programming conference* in Allerton Park, Illinois, September 4-6, 1996.
15. "Acceptor and Connector: Design Patterns for Initializing Network Services," Proceedings of the EuroPLoP '96 conference, Kloster Irsee, Germany, July 10-14, 1996.
16. "Measuring the Performance of Communication Middleware on High-Speed Networks," Proceedings of SIGCOMM '96, ACM, San Francisco, August 28-30th, 1996.
17. "Design and Performance of an Object-Oriented Framework for High-Speed Electronic Medical Imaging," Proceedings of the 2nd Conference on Object-Oriented Technologies and Systems (COOTS), USENIX, Toronto, June 18-22, 1996.
18. "A Family of Design Patterns For Flexibly Configuring Network Services in Distributed Systems," Proceedings of the International Conference on Configurable Distributed Systems, IEEE, Annapolis, Maryland, May 6-8, 1996.
19. "Using Design Patterns to Develop High-Performance Object-Oriented Communication Software Frameworks," Proceedings of the 8th Annual Software Technology Conference, Salt Lake City, Utah, April 21-26, 1996.
20. "An Object-Oriented Framework for High-Performance Electronic Medical Imaging," Proceedings of the *Very High Resolution and Quality Imaging* mini-conference at the Symposium on Electronic Imaging in the International Symposia Photonics West 1996, SPIE, San Jose, California USA, January 27 - February 2, 1996.
21. "Half-Sync/Half-Async: A Pattern for Efficient and Well-structured Concurrent I/O," *Proceedings of the 2nd Pattern Languages of Programs Conference* Monticello, Illinois, September 6-8, 1995.

22. "Acceptor and Connector: Design Patterns for Actively and Passively Initializing Network Services." Workshop on Pattern Languages of Object-Oriented Programs at ECOOP '95, August 7 – 1, 1995, Aarhus, Denmark.
23. "Object-Oriented Components for High-speed Network Programming," *Proceedings of the Conference on Object-Oriented Technologies (COOTS)*, USENIX, June 26-29, 1995 Monterey, California, USA, pp. 21–38.
24. "Experience Using Design Patterns to Evolve Communication Software Across Diverse OS Platforms," *Proceedings of the 9th European Conference on Object-Oriented Programming (ECOOP)*, ACM, Aarhus, Denmark, August, 1995,.
25. "Measuring the Performance of Parallel Message-based Process Architectures," *Proceedings of the INFOCOM Conference on Computer Communications*, IEEE, Boston, MA, April, 1995, pp. 624–633.
26. "High-Performance Event Filtering for Dynamic Multi-point Applications," Proceedings of the 1st Workshop on High Performance Protocol Architectures (HIPPARCH), INRIA, Sophia Antipolis, France, December, 1994, p 1–8.
27. "Flexible Configuration of High-Performance Object-Oriented Distributed Communication Systems," *9th OOPSLA Conference, invited paper to the Workshop on Flexibility in Systems Software*, ACM, Portland, Oregon, October, 1994, pp. 1–4.
28. "Performance Experiments on Alternative Methods for Structuring Active Objects in High-Performance Parallel Communication Systems," *9th OOPSLA Conference, poster session*, ACM, Portland, Oregon, October, 1994, pp. 1–12.
29. "Measuring the Impact of Alternative Parallel Process Architectures on Communication Subsystem Performance," *Proceedings of the Proceedings of the 4th International Workshop on Protocols for High-Speed Networks*, IFIP, Vancouver, British Columbia, August, 1994, pp. 103–118.
30. "Reactor: An Object Behavioral Pattern for Concurrent Event Demultiplexing and Dispatching," *Proceedings of the 1st Annual Conference on the Pattern Languages of Programs*, Monticello, Illinois, August, 1994, pp. 1–10.
31. "Experiences with an Object-Oriented Architecture for Developing Dynamically Extensible Network Management Software," *Proceedings of the Globecom Conference*, IEEE, San Francisco, California, November, 1994, pp. 1–7.
32. "Configuring Function-based Communication Protocols for Distributed Applications," *Proceedings of the 8th International Working Conference on Upper Layer Protocols, Architectures, and Applications*, IFIP, Barcelona, Spain, June 1-3, 1994, pp. 361–376.
33. "The ADAPTIVE Service Executive: An Object-Oriented Architecture for Configuring Concurrent Distributed Communication Systems," *Proceedings of the 8th International Working Conference on Upper Layer Protocols, Architectures, and Applications*, IFIP, Barcelona, Spain, June 1-3, 1994, pp. 163–178.

34. "ASX: An Object-Oriented Framework for Developing Distributed Applications," *Proceedings of the 6th C++ Conference*, USENIX, Cambridge, Massachusetts, April, 1994, pp. 200–220.
35. "The Service Configurator Framework: An Extensible Architecture for Dynamically Configuring Concurrent, Multi-service Network Daemons," *Proceedings of the 2nd International Workshop on Configurable Distributed Systems*, IEEE, Pittsburgh, PA, March 21-23, 1994, pp. 190–201.
36. "Tools for Generating Application-Tailored Multimedia Protocols on Heterogeneous Multi-Processor Platforms," *Proceedings of the 2nd Workshop on High-Performance Communications Subsystems*, IEEE, Williamsburg, Virginia, September 1-3, 1993, pp. 1–7.
37. "A Framework for Developing and Experimenting with Parallel Process Architectures to Support High-Performance Transport Systems," *Proceedings of the 2nd Workshop on High-Performance Communications Subsystems*, IEEE, Williamsburg, Virginia, September 1-3, 1993, pp. 1–8.
38. "ADAPTIVE: a Framework for Experimenting with High-Performance Transport System Process Architectures," *Proceedings of the 2nd International Conference on Computer Communications and Networks*, ISCA, San Diego, California, June 28-30, 1993, pp. 1–8.
39. "ADAPTIVE: A Flexible and Adaptive Transport System Architecture to Support Lightweight Protocols for Multimedia Applications on High-Speed Networks," *Proceedings of the 1st Symposium on High Performance Distributed Computing*, IEEE, Syracuse, New York, September 9-11, 1992, pp. 174–186.
40. "GPERF: A Perfect Hash Function Generator," *Proceedings of the 2nd C++ Conference*, USENIX, San Francisco, California, April 9-11, 1990, pp. 87–102.

Invited Talks

1. "Towards Technical Reference Frameworks to Support Open System Architecture Initiatives," Office of the Secretary of Defense (OSD) System of Systems Engineering Collaborators Information Exchange, December 15th 2015.
2. "Enterprise System of Systems Engineering (SoSE) Integration and Innovation," presentation at the US Marine Corp Business Management Association meeting, Quantico, VA, December 10th, 2015.
3. "An Architecture Grand Challenge: DOD's push for Open Systems Architecture," panel presentation at the Software Solutions Conference, Crystal City, VA, November 17th, 2015.
4. "Elastic Software Infrastructure to Support the Industrial Internet," the Siemens CPS Workshop, Munich, Germany, September 29th, 2015.
5. "An Overview of Mobile and mHealth Activities at ISIS and Vandy EECS," Patient Engagement Emerging Technologies, Vanderbilt University, August 10, 2015.

6. "Mobile Cloud Computing with Android," Google I/O, Mercury Intermedia Systems, May 28th, 2015.
7. "An Architecture Grand Challenge: DOD's push for Open Systems Architecture," panel presentation at the SATURN 2015 Conference, Baltimore, MD, April 27th, 2015.
8. "'Elastic Software Infrastructure to Support Computing Clouds for Cyber-Physical Systems," Distinguish Lecture, Texas A&M, April 27th, 2015.
9. "Elastic Software Infrastructure to Support Computing Clouds for Cyber-Physical Systems", Boeing Distinguished Researcher And Scholar Seminar (B-DRASS) series, March 20th, Huntingon Beach, CA.
10. "Elastic Software Infrastructure to Support Computing Clouds for Cyber-Physical Systems," Distinguished Lecture, University of California, Irvine, February 27th, 2015.
11. "Elastic Software Infrastructure to Support Computing Clouds for Cyber-Physical Systems," Varian, Palo Alto, CA, January 15th, 2015.
12. "Keeping an Unfair Advantage in a Globalized and Commoditized World," Open Architecture Summit, Washington DC, November 4th, 2014.
13. "Proposal for a Professional Masters degree in Computer Science," invited talk at Vanderbilt University School of Engineering's Board of Visitor's meeting, October 10th, 2014.
14. "The Past, Present, and Future of Open System Architecture Initiatives," invited keynote at the Future Airborne Capabilities Environment meeting, Nashville, TN, September 24th.
15. "Future Proofing Research and Development Investments in a Globalized, Commoditized World," Boeing Technical Excellence Conference, May 20th, 2014, St. Louis, MO.
16. "Elastic Software Infrastructure to Support the Computing Clouds for Cyber-Physical Systems (CC4CPS)," Securboration Conference, November 12th, 2013, Melbourne, Florida.
17. "The Importance of Automated Testing in Open Systems Architecture Initiatives," Open Architecture Summit, November 12th, 2013, Washington DC.
18. Conference on Cloud and Mobile Computing, Siemens Corporate Research, Princeton, NJ, November 5th, 2013.
19. "Overview of the Technology Entrepreneurship Task Force," Innovation, Imagination, and Introductions: A Conversation with Entrepreneurs, Vanderbilt University, October 24th, 2013.
20. "Producing and Delivering a Coursera MOOC on Pattern-Oriented Software Architecture for Concurrent and Networked Software," Vanderbilt University's Faculty Senate committee on Strategic Planning and Academic Freedom, October 23rd, 2013.
21. "Elastic Software Infrastructure to Support the Industrial Internet," RTI Webinar series, October 23rd, 2013.
22. "The Importance of Applying Agility to DoD Software Initiatives," IEEE Computer Society Lockheed Martin webinar series, October 10th, 2013.

23. "Technology Entrepreneurship Task force: Charter and Progress Update," Vanderbilt University School of Engineering Board of Visitors meeting, October 4th, 2013.
24. "Stochastic Hybrid Systems Modeling and Middleware-enabled DDDAS for Next-generation USAF Combat Systems," AFOSR DDDAS PI meeting, Arlington, VA, October 1st, 2013.
25. "Producing and Delivering a Coursera MOOC on Pattern-Oriented Software Architecture for Concurrent and Networked Software," WithIT seminar, Vanderbilt University, September 12th, 2013.
26. "Applying Agility to the US Department of Defense Common Operating Platform Environment Initiatives," Interoperable Open Architecture conference, Washington DC, September 11th, 2013.
27. "Software Infrastructure Support of Computing Clouds for Cyber-Physical Systems," invited talk at Real-Time Innovations, July 31st, 2013, Sunnyvale, California.
28. "Introduction to the Institute for Software Integrated Systems," Nashville Entrepreneur Center, July 15th, 2013.
29. "Surviving the Coursera Digital Learning Experience," Coursera-in-TN Conference, Vanderbilt University, Nashville, TN, June 24th, 2013.
30. "Quo Vadis ISORC? ," Panel presentation at ISORC 2013 Conference, June 19th, 2013, Paderborn, Germany.
31. "Software Infrastructure Support of Computing Clouds for Cyber-Physical Systems," invited keynote for ISORC 2013 Conference, June 19th, 2013, Paderborn, Germany.
32. "Towards Programming Models and Paradigms for Computing Clouds that Support Cyber-Physical Systems," NSF Workshop on Computing Clouds for Cyber-Physical Systems, March 15th, 2013, Ballston, VA.
33. "Built to Last: Planning Your Career as an Engineer," STEM contest on Securing Cyber Space, Brentwood High School, March 9th, 2013, Nashville, TN.
34. "Experience with Digital Learning and MOOCs at Vanderbilt," Nashville, TN, Feb 22nd, 2013.
35. "Software Design: Is It Really Better to Look Good Than to Feel Good? ," World IA Day, Nashville, TN, Feb 9th, 2013.
36. "Pattern-Oriented Software Architectures: Patterns and Frameworks for Concurrent and Networked Software," PhreakNIC 2012, Murfreesboro, TN, November 9th, 2012.
37. "Applying Agility to the US Department of Defence Common Operating Platform Environment Initiatives," Interoperable Open Architecture 2012, 29 - 31 October, 2012, London, UK.
38. "Open System Architectures: Challenges and Success Drivers," OA Summit conference, Washington, DC, October 18th, 2012.
39. "Dependable Computing Clouds for Cyber-Physical Systems," Dependability Issues in Cloud Computing Workshop, October 11th, 2012, Irvine, CA.

40. "Computing Clouds for Cyber-Physical Systems," Reliable Cloud Infrastructure for CPS Applications Workshop, October 8th, 2012, Irvine, CA.
41. "Common Operating Platform Environments: Challenges and Success Drivers," Navy Open Systems Architecture workshop, Ballston, VA, September 27th, 2012.
42. "Meeting the Challenges of Enterprise Distributed Real-time and Embedded Systems," talk for Honeywell Aerospace, September 21, 2012.
43. "Architecture-Led Iterative and Incremental Development for Common Operating Platform Environments," NITRD Software Design and Productivity meeting, National Coordination Office, Ballston, VA, July 13th, 2012.
44. "Cyber-physical multi-core Optimization for Resource and cachE effectS," Software-Intensive Systems Producibility workshop, Arlington VA, June 5th, 2012.
45. "Applying Agility to DoD Common Operating Platform Environment Initiatives", SEI Agile Research Forum, May 22nd, 2012.
46. "Meeting the Challenges of Enterprise Distributed Real-time and Embedded Systems," keynote talk at the SATURN Conference 2012 May 7-11, 2012, St. Petersburg, FL.
47. "Reflections on 20 Years of Architecture for Distributed Real-time and Embedded Systems," SATURN Conference 2012 May 7-11, 2012, St. Petersburg, FL.
48. "US Naval Open Systems Architecture Strategy," SATURN Conference 2012 May 7-11, 2012, St. Petersburg, FL.
49. "Towards Open Systems Architectures for Distributed Real-time and Embedded Systems," The Center for Embedded Systems for Critical Applications, Annual Workshop, Virginia Tech, Blackburg, VA April 21st, 2012.
50. "Overview of the SEI Strategic Research Plan," ASD(R&E) Annual Program Review, December 7th, 2011, Pittsburgh, PA.
51. "Overview of the SEI Strategic Research Plan," Acquisition Support Program meeting, November 16th, 2011, Pittsburgh, PA.
52. "Conducting Leading-Edge Software R&D in a Globalized, Commoditized World," NITRD Software Design and Productivity meeting, National Coordination Office, Ballston, VA, November 3rd, 2011.
53. "A Technical Assessment of Open Architecture Systems for Military Use," Interoperable Open Architecture, 26th-28th October 2011, London, UK.
54. "Conducting Leading-Edge Software R&D in a Globalized, Commoditized World," Technovation 2011, Carnegie Mellon University, September 29th, 2011.
55. "CTO Report," SEI Board of Visitors Meeting, Arlington, VA, September 27th, 2011.
56. "Overview of the SEI Strategic Research Plan," Joint Advisory Committee Meeting, Arlington, VA, September 26th, 2011.
57. "Successful Development Efforts: Standards, People, & Culture: The Enterprise Perspective," Software Assurance (SwA) Forum, September 16th, 2011, Arlington, VA.

58. "Ultra-Large-Scale (ULS) Cyberphysical Systems and Their Impact on Technology and Society," University of Salzburg, June 30th, 2011, Salzburg, Austria.
59. "Ultra-Large-Scale (ULS) Cyberphysical Systems and Their Impact on Technology and Society," ARTEMIS conference, June 29th, 2011, Linz, Austria.
60. "Ultra-Large-Scale Systems and Their Impact on the DoD," Systems and Software Technology Conference Committee, keynote presentation at the 23rd Systems and Software Technology Conference, May 16-19, 2011, Salt Lake City, Utah.
61. "Ultra-Large Scale Systems and their Impact on Technology and Society," keynote presentation at the International Symposium on Object-Oriented Real-time Distributed Computing</A> (ISORC), Newport Beach, CA, March 29th, 2011.
62. "Software-reliant Systems Research at the Software Engineering Institute," Raytheon, Sudbury, MA, March 10, 2011.
63. "Review of COE Practices," US Army Senior Leadership Education Program, Pittsburgh, PA, January 20th, 2011.
64. "Software Producibility for Defense," US Army Senior Leadership Education Program, Pittsburgh, PA, January 18th, 2011.
65. "SEI Research: The Shape of Things to Come," ASP Meeting, Software Engineering Institute, Pittsburgh, PA, December 9th, 2010.
66. "R&D at ASP," ASP Air Force Training Day, Software Engineering Institute, Pittsburgh, PA, December 9th, 2010.
67. "Software-reliant Systems Research at the Software Engineering Institute," Siements Corporate Research, Princeton, NJ, November 22nd, 2010.
68. "Taming the Complexity of Software-Reliant Systems," Software Engineering Process Group conference, Colombia, South America, November 11th, 2010.
69. "SEI Technical Presentations," Joint Advisory Committee Meeting, Arlington, VA, October 26th, 2010.
70. "SEI Research: The Shape of Things to Come," ASP Meeting, Software Engineering Institute, Pittsburgh, PA, October 20th, 2010.
71. "SEI Research: The Shape of Things to Come," SEPM Meeting, Software Engineering Institute, Pittsburgh, PA, October 19th, 2010.
72. "Strategic Directions for Research at the SEI," RTSS Offsite Meeting, Pittsburgh, PA, October 12th, 2010.
73. "The World is Flat and What You Can Do About It," Family Weekend, October 9th, 2010, Vanderbilt University.
74. "SEI Research: The Shape of Things to Come," SEI Board of Visitor's Meeting, Arlington, VA, September 28th, 2010.
75. "SEI Research: The Shape of Things to Come," PD&T Meeting, Software Engineering Institute, Pittsburgh, PA, September 20th, 2010.
76. "Introduction and Initial Thoughts," RTSS Meeting, Software Engineering Institute, Pittsburgh, PA, August 19th, 2010.

77. "The Impact of Ultra-Large-Scale Systems on DoD Operations," Congressional R&D Caucus, Rayburn Building, Washington DC, January 19th, 2010.
78. "The World is Flat and What You Can Do About It," Explorers meeting, January 12th, 2010, Vanderbilt University.
79. "Expectations for University - Industry Collaborative Research in CPS," Computing Community Consortium Workshop on New Forms of Industry-Academy Partnerships in CPS Research, George Mason University, May 19th, 2009.
80. "How Good is Your SOA? ", Panel presentation at the AFRL QED PI meeting, April 28th, 2009, Washington DC.
81. "The World is Flat and What You Can Do About It," ES 140, Computer Science module, October 31st, 2008, Vanderbilt University.
82. "Meeting the Challenges of Ultra-Large-Scale Distributed Real-time and Embedded Systems with QoS-enabled Middleware and Model-Driven Engineering," Panel on Growing and Sustaining Ultra Large Scale (ULS) Systems, OOPSLA 2008, Nashville TN, October 21-23 2008.
83. "The World is Flat and What You Can Do About It," Family Weekend Faculty Lecture, Vanderbilt University, October 3rd, 2008.
84. "The World is Flat and What You Can Do About It," Senior Design Seminar, Vanderbilt University, September 17th, 2008.
85. "The World is Flat and What You Can Do About It," CS WithIT Seminar, Vanderbilt University, September 11th, 2008.
86. "The Managed Motorway: Real-time Vehicle Scheduling - A Research Agenda," Qualcomm, July 28th, 2008, San Diego, CA.
87. "Meeting the Challenges of Mission-Critical Distributed Event-Based Systems with QoS-enabled Middleware and Model-Driven Engineering," 2nd International Conference on Distributed Event-Based Systems (DEBS), Rome Italy, July 2-4, 2008.
88. "Meeting the Challenges of Distributed Real-time and Embedded Systems with QoS-enabled Middleware and Model-Driven Engineering," SPAWAR, April 29th, 2008.
89. "Meeting the Challenges of Distributed Real-time and Embedded Systems with QoS-enabled Middleware and Model-Driven Engineering," Northrop Grumman, Boulder Colorado, April 25th, 2008.
90. "Experimentation Environment for QED," AFRL Information Management PI Meeting, April 16 2008, Georgetown, Washington, DC.
91. "Adaptive System Infrastructure for Ultra-Large-Scale Systems," SMART Conference, Carnegie Mellon University, March 6th, 2008.
92. "Experimentation Environment for QED", Air Force Research Lab, Rome, NY, March 4th, 2008.
93. "Ultra-Large-Scale (ULS) Systems and their Impact on Technology and Society," Clemson University, January 31st, 2008.
94. "Meeting the Challenges of Ultra-Large-Scale Distributed Real-time and Embedded Systems with QoS-enabled Middleware and Model-Driven

Engineering, invited keynote talk at Middleware 2007, Irvine, CA, November 29th, 2007.

95. "The World is Flat and What You Can Do About It," Senior Design Seminar, Vanderbilt University, November 14th, 2007.
96. "Technology Candidates for QED," AFRL retreat, Minnowbrook, NY, October 23, 2007.
97. "Overview of ISIS and Proposed IU/CRC R&D Projects," Crystal City, VA, October 19th, 2007.
98. The Future of CORBA for Distributed Real-time and Embedded Systems, International Conference on Accelerator and Large Experimental Physics Control Systems, October 17, 2007, Knoxville, TN.
99. "AF-TRUST: Project Overview," Air Force Science Advisory Board review, Rome, NY, October 15th, 2007.
100. "Meeting the Challenges of Distributed Real-time and Embedded Systems with Product-Line Architectures," August 1st, 2007, Trinity College, Dublin, Ireland.
101. "Model Driven Engineering of Product-Line Architectures for Distributed Real-time and Embedded Systems," July 5th, 2007, University of Limerick, Ireland.
102. "Meeting the Challenges of Mission-Critical Systems with Middleware and Model Driven Engineering", OMG Technical Meeting, June 27, 2007, Brussels, Belgium.
103. Meeting the Challenges of Ultra-Large-Scale Distributed Real-time and Embedded Systems with Model-Driven Engineering, June 19, 2007, Trinity College, Dublin.
104. Strategic Technology Positioning, PrismTechnologies "Middleware Fest", June 14, 2007, Newcastle, UK.
105. "Hurdles for Wireless Communication Systems R&D and Some Ways to Overcome Them," OSD Workshop on Wireless Communication Systems, Rosslyn, VA, May 22nd, 2007.
106. "The World is Flat from a Computer Scientists Point of View," Vanderbilt University Commencement talk, May 10th, 2007.
107. Meeting the Challenges of Ultra-Large-Scale Distributed Real-time and Embedded Systems, invited keynote at the the 10th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing, May 7-9, 2007, Santorini Island, Greece.
108. "Enhanced QoS for the GIG," AFRL JBI PI meeting, Georgetown, DC, April 24, 2007.
109. "Meeting the Challenges of Ultra-Large-Scale Distributed Real-time and Embedded Systems," Invited keynote at the 15th International Workshop on Parallel and Distributed Real-Time Systems (WDPRTS), March 26-27, 2007, Long Beach, California.
110. "The CORBA C++ Mapping: Beyond Repair? ," OMG Meeting, San Diego, CA, March 27th, 2007.

111. "Meeting the Challenges of Ultra-Large-Scale Systems via Model-Driven Engineering," Distinguished Lecturer Series, Florida International University, Miami, Florida, Feb 2, 2007.
112. Model Driven Engineering and QoS-enabled Component Middleware for DRE Systems, Invited talk at the European Space Agency Operations Center, Darmstadt, Germany, Wednesday January 24, 2007.
113. "Software Wind Tunnel (SWiT) Concept of Operations and System Architecture", AFRL Software and Systems Test Track workshop, Arlington, VA, January 19, 2007.
114. "Latest Breakthroughs in SDR Software Development Using Model Driven Technologies," Rockwell Collins, Cedar Rapids, IA, December 14th, 2006.
115. "Educating the DoD Workforce in a Flat World," 2006 Raytheon Integrated Defense Systems' SW Engr. Directorate Off-Site Meeting, New Castle, New Hampshire, December 7, 2006.
116. "The Ultra Challenge: Software Systems Beyond Big," panelist at OOPSLA 2006, October, 2006, Portland, OR.
117. "Software Wind Tunnel (SWiT) Architecture," AFRL Software and Systems Test Track Workshop, Cherry Hill, NJ, October 2nd, 2006.
118. "The World is Flat and What You Can Do About it," Vanderbilt University, September 12th, 2006.
119. "The World is Flat and What You Can Do About it," Vanderbilt University, September 8th, 2006.
120. "Meeting the Challenges of Ultra-Large-Scale Systems via Model-Driven Engineering," Network-Centric Operations Industry Consortium, Reston, VA, August 2nd 2006.
121. Model Driven Architecture Roundtable, invited panelist at the Software Engineering Institute, Pittsburgh, PA, June 1st, 2006.
122. "Enhanced QoS for the GIG," AFRL JBI PI meeting, Tysons Corner, VA, April 11, 2006.
123. "Model Driven Engineering for Distributed Real-time and Embedded Systems," Distinguished Lecturer Series talk at Colorado State University, Ft. Collins, CO, April 10, 2006.
124. "Win-Win Partnership of Academia and Industry: Why Should We Care? Where Is Our Common Future? " invited panelist at the 12th IEEE Real-Time and Embedded Technology and Applications Symposium April 6, 2006, San Jose, California.
125. "Meeting the Challenges of Ultra-Large-Scale Real-time Systems," invited keynote at the IEEE Real-Time and Embedded Technology and Applications Symposium April 5, 2006, San Jose, California.
126. "Model-driven Development for Distributed Real-time and Embedded Systems," ACM Meeting at Middle Tennessee State University, March 7th, 2006.
127. "Real-time, Scalable, and Secure Information Management for the GIG," Science Advisory Board Meeting, Rome, NY, November 16th, 2005.

128. "Real-time, Scalable, and Secure Information Management for the GIG," Airforce Research Lab, Rome, NY, November 3rd, 2005.
129. "Model-driven Development for Distributed Real-time and Embedded Systems," Distinguished Speaker Talk at BBN Technologies, Cambridge, MA, October 27, 2005.
130. "Challenges and Research Areas for QoS-enabled Information Management in Tactical Systems of Systems," AFRL Minnowbrook Workshop, Adirondack Mountains, NY, October 21st, 2005.
131. "Model-driven Development for Distributed Real-time and Embedded Systems," Invited keynote at MODELS 2005, ACM/IEEE 8th International Conference on Model Driven Engineering Languages and Systems, Half Moon Resort, Montego Bay, Jamaica, October 5-7, 2005.
132. "The World is Flat and What You Can Do About it," CS WithIT Seminar, Vanderbilt University, September 22, 2005.
133. "Why Software Reuse has Failed and How to Make it Work for You," Motorola 2005, Symposium on Software, Systems, and Simulation, Schaumburg, IL, September 16th, 2005.
134. "Pattern-Oriented Software Architecture," 12th Pattern Language of Programming Conference, Allerton Park, Illinois, September 7-10, 2005.
135. "Model-Driven Development of Distributed Real-time and Embedded Systems," 12th Pattern Language of Programming Conference, Allerton Park, Illinois, September 7-10, 2005.
136. "Model-driven Development for Distributed Real-time and Embedded Systems," Siemens Corporate Research, Princeton, NJ, August 26th.
137. "Model-driven QoS Provisioning for Real-time CORBA and CCM DRE Systems," 6th OMG Real-time/Embedded CORBA workshop, Washington DC, July 11–14, 2005.
138. "A Proposed R&D Agenda for the Software Technology Laboratory," Lockheed Martin Advanced Technology Lab, Cherry Hill, NJ, June 28th, 2005.
139. "Model-Driven Development of Product-Line Architectures for DRE Systems," 11th Siemens Software Architecture Improvement Group (SAIG), Buffalo Grove, IL June 22, 2005.
140. "Business Drives for Platforms," panel at the 11th Siemens Software Architecture Improvement Group (SAIG), Buffalo Grove, IL June 22, 2005.
141. "Model Driven Development for Distributed Real-time and Embedded Systems," Lockheed Martin Advanced Technology Lab, Cherry Hill, NJ, June 15th, 2005.
142. "Approaches for Supporting Real-time QoS in JBI," JBI PI Meeting, Washington DC, May 24th, 2005.
143. "Overcoming Hurdles of Software Producibility," OSD, Software Producibility Workshop, Arlington, VA, May 18, 2005.
144. "Overview of Multi-Level Resource Management in ARMS," Fermilab, Chicago, IL, April 12th, 2005.
145. "Model Driven Middleware for Distributed Real-time and Embedded Systems," University of Southern Alabama, April 8, 2005.

146. "Model-Driven Development of Distributed Real-time and Embedded Systems," UAV Battlelab, Indian Springs, NV, February 10th, 2005.
147. "The Future of Software and Systems Engineering," IEEE Meeting, Vanderbilt University, February 8th, 2005.
148. Model Driven Development of Distributed Real-time and Embedded Systems, panel at the OOP conference, Munich, Germany, January 27, 2005.
149. "Product-line Architecture Technologies for Distributed Real-time and Embedded Systems, Lockheed Martin, Moorestown, NJ, November 11, 2004.
150. "Model Driven Development of Distributed Real-time and Embedded Systems," invited panelist in the "Generative Programming: Past, Present, and Future," at the 3rd ACM International Conference on Generative Programming and Component Engineering, Vancouver, CA, October 24th 2004.
151. "Developing Combat Systems with Component Middleware and Models," Lockheed Martin, Moorestown, NJ, October 22, 2004.
152. "Model Driven Development of Distributed Real-time and Embedded Systems," Lockheed Martin Advanced Technology Lab, Cherry Hill, NJ, October 21, 2004.
153. "Model Driven Development of Distributed Real-time and Embedded Systems," Lockheed Martin Missile and Fire Control, Dallas, TX, October 13, 2004.
154. "Design of ARMS MLRM Components: CCM Based Design for Dynamic Resource Management," DARPA ARMS Technical Interchange Meeting, Plymouth, RI, October 7, 2004.
155. "Model Driven Middleware for Component-based Distributed Systems," keynote for the The 8th International IEEE Enterprise Distributed Object Computing Conference, Monterey, California, September 22, 2004.
156. "Systems Science Challenge Area," TRUST NSF Science and Technology Review, UC Berkeley, September 12, 2004.
157. "Model Driven Development for Distributed Real-time and Embedded Systems," Lockheed Martin, Eagan, MN, August 31st, 2004.
158. "Model Driven Computing for Distributed Real-time and Embedded Systems," Telcordia, Piscataway, NJ, August 10th, 2004.
159. "Model Driven Computing for Distributed Real-time and Embedded Systems," Raytheon, Portsmouth, RI, August 9th, 2004.
160. "Distributed Object Computing with CORBA," Raytheon, Portsmouth, RI, August 9th, 2004.
161. "Model Driven Development of Distributed Real-time and Embedded Systems," Raytheon, Ft. Wayne, IN, July 27th, 2004.
162. "Model Driven Middleware for Distributed Real-time and Embedded Systems," panelist at the 5th OMG Real-time and Embedded Middleware Workshop, Reston, VA 2004.
163. "The Role of Open Standards,Open-Source Development, and Different Development Models and Processes on Industrializing Software," ARO Workshop on Software Reliability for FCS, Vanderbilt University, Nashville, Tennessee, May 18-19, 2004.

164. "Model Driven Middleware for Distributed Real-time and Embedded Systems," Keynote talk for the SIGS Software Engineering Today conference in Zurich, Switzerland, May 4-5, 2004.
165. "Model-Driven Development of Distributed Real-time and Embedded Systems," 10th Siemens Software Architecture Improvement Group (SAIG), Vienna, Austria, April 20-24.
166. "Adaptive and Reflective Middleware for Distributed, Real-time, and Embedded Systems," Purdue University, West Lafayette, Indiana, April 6, 2004.
167. "Model Driven Middleware for Distributed Real-time and Embedded Systems," *Technologies That Will Change the World* session at the Southeastern Software Engineering Conference, Huntsville, Alabama, March 30th, 2004.
168. "Advances in COTS Middleware for Distributed Real-time and Embedded Systems," Keynote for the International Conference on COTS-Based Software Systems (ICCBSS) 2004 in Redondo Beach, February 2-4, 2004.
169. Composable Middleware Components for High Confidence Network Embedded Systems, University of California, Berkeley, December 4th, 2003.
170. "Model Driven Middleware," TechConnect 2003, St. Louis, MO, October 1st, 2003.
171. "Advances in Model Driven Middleware for Distributed Real-time and Embedded Systems," the Model Integrated Computing PSIG meeting at the OMG Technical Meeting, September 10, 2003, Boston, MA.
172. Invited panelist for the "Research on DRE Systems" panel at the OMG Real-time Middleware Workshop, July 16, 2003, Arlington, VA.
173. "Advances in Model Driven Middleware for Distributed Real-time and Embedded Systems," the OMG Real-time Middleware Workshop, July 15, 2003, Arlington, VA.
174. Organizer and presenter for a panel on "Advances in Large-scale Distributed Real-time and Embedded Systems" at the 9th IEEE Real-time/Embedded Technology and Applications Symposium (RTAS), May 27-30, 2003, Washington, DC.
175. "Managing Project Risk for Combat Systems," The Southeastern Software Engineering Conference, Huntsville, Alabama, April 1st, 2003.
176. "Distributed Real-time and Embedded Systems at DARPA," OMG Workshop on Super Distributed Objects, Washington DC, Monday, November 18, 2002.
177. "Adaptive and Reflective Middleware for Distributed Real-time Systems," Workshop on High Performance, Fault Adaptive, Large Scale Real-time Systems, Vanderbilt University, November 14, 2002.
178. Invited panelist on "Objects and Real-time Systems" OOPSLA '02, Seattle, WA, November 8, 2002.
179. "An Overview of ACE+TAO," Boeing, Seattle, November 8th, 2002.
180. "Pattern-Oriented Software Architecture," Amazon, Seattle, WA, November 6th, 2002.
181. "Using Real-time CORBA Effectively: Patterns and Principles," CORBA Controls Workshop, Grenoble, France, October, 9th, 2002.

182. "Adaptive and Reflective Middleware for Distributed Real-time and Embedded Systems," EMSOFT 2002: Second Workshop on Embedded Software, Grenoble, France, October, 7–9th, 2002.
183. "Designing the Future of Embedded Systems at DARPA IXO," Keynote talk at the 6th Annual Workshop on High-Performance Embedded Computing (HPEC), September 25, Boston, MA.
184. "Open Distributed Computing Platforms," NSF/OSTP Workshop on Information Technology Research for Critical Infrastructure Protection, Lansdowne, VA, September 20th, 2002.
185. "Real-time Object-Oriented Middleware," Distributed Common Ground/Surface System Technical Review Group meeting, Mclean VA, September 19th, 2002.
186. "Research Advances in Middleware for Distributed, Real-time, and Embedded Systems," Computer Communications stream of the 17th IFIP World Computer Congress, Montreal, Canada, August 25-30, 2002.
187. "DARPA Thrusts in Embedded Computing," Mercury Computer Systems, Tyngsboro, MA, July 25th, 2002.
188. "Adaptive and Reflective Middleware for Distributed, Real-time, and Embedded Combat Systems," Boeing Space and Missile Systems, Anaheim, CA, July 9, 2002.
189. "Annual Report on Software Design and Productivity Coordinating Group," Interagency Working Group, ITR&D Spring Planning Meeting, NSF, Ballston, VA, May 10, 2002.
190. "Real-time CORBA Standardization: Past, Present, and Future," panelist in the "Standards Movements in Object-oriented Real-time Computing" panel at the ISORC 2002 Conference, Washington, DC, April 30, 2002.
191. "Towards Adaptive and Reflective Middleware for Distributed Real-time Embedded Systems," Moderator of the *Distributed, Real-time, and Embedded Middleware for Network-Centric Combat Systems* panel at the Software Technology Conference (STC) in Salt Lake City, Utah, April 29, 2002.
192. "Applying Architectural Patterns to Address Key Challenges of Distributed Software," Siemens Architecture Interworking Group, Chicago, IL, April 24, 2002.
193. "Towards Adaptive and Reflective Middleware for Distributed Real-time and Embedded Systems," Space and Missile Defense Command, Huntsville, AL, April 22, 2002.
194. "How to Maintain Superiority in the Face of the Commoditization of IT," tutorial at the UCI CEO Roundtable, Maui, Hawaii, April 12, 2002.
195. "Transformation or Transmogrification? Surviving the Commoditization of IT," panelist at the UCI CEO Roundtable, Maui, Hawaii, April 11, 2002.
196. "Patterns and Principles of Mission-critical Middleware," Henry Samueli School of Engineering Research Review, University of California, Irvine, March 14th, 2002.
197. "DARPA: an Agency Overview," CRA Academic Careers Workshop, Arlington, Virginia, February 10 - 12, 2002.

198. "Towards Adaptive and Reflective Middleware for Distributed, Real-time, and Embedded Systems," Electrical Engineering and Computer Science Department, Vanderbilt University, January 28th, 2002.
199. "Protecting Critical Cyber Infrastructure from Asymmetric Threats," panelist at the 7th IEEE Workshop on Object-oriented Real-time Dependable Systems, San Diego, CA, January 10, 2002.
200. "The Researcher's Dilemma: When Technology Success Causes Great Communities to Fail (at Mission-oriented R&D Agencies)," Software Design and Productivity Coordinating Group Workshop on New Visions for Software Design and Productivity: Research and Applications, Nashville, TN, December 13-14, 2001.
201. "Towards Adaptive and Reflective Middleware for Mission-Critical Systems," Computer Science Department, College of William and Mary, September 7th, 2001.
202. "Adaptive and Reflective Middleware Systems," Lockheed Martin, Moorestown, NJ, August 21st, 2001.
203. "Adaptive and Reflective Middleware Systems," United Technology Research Center, Hartford, Connecticut, June 28th, 2001.
204. "Adaptive and Reflective Middleware Systems," Raytheon Annual Processing Systems Technology Network (PSTN) Symposium, Lexington, MA, June 20th, 2001.
205. Invited presenter for the Vendors' Panel at the OMG 2nd Workshop on Real-time and Embedded Distributed Object Computing, June 4-7, 2001.
206. "Towards Pattern Languages and QoS-enabled Middleware for Distributed Real-time and Embedded Systems," DARPA ITO workshop on Embedded Software, Lake Tahoe, NV, October 8-10, 2001.
207. "TAO, CORBA, and the HLA/RTI", Keynote talk at the Fifth IEEE International Workshop on Distributed Simulation and Real Time Applications Cincinnati, Ohio, USA August 13-15, 2001.
208. "Patterns and Principles of Middleware for Distributed Real-time and Embedded Systems," Raytheon, Sudbury, March 29th, 2001.
209. "Adaptive and Reflective Middleware Systems," Distinguished Lecture at Florida Atlantic University, Boca Raton, FL, March 1st, 2001.
210. "Adaptive and Reflective Middleware for Mission-Critical Distributed and Embedded Systems," University of Alabama, Birmingham, AL, January 31st, 2001.
211. "Adaptive and Reflective Middleware for Mission-Critical Distributed and Embedded Systems," Telcordia, Morristown, NJ, November 20th, 2000.
212. "Adaptive and Reflective Middleware for Mission-Critical Distributed and Embedded Systems," George Mason University, Fairfax, VA, November 20th, 2000.
213. "Adaptive and Reflective Middleware for Mission-Critical Distributed and Embedded Systems," Lucent CORBA Forum, Naperville, IL, November 17th, 2000.

214. "Putting an ORB on a Diet," Session on *Performance and QoS of Embedded CORBA ORBs* at the OMG's Workshop on Embedded Object-Based Systems, January 17-19, 2001.
215. "Adaptive and Reflective Middleware Systems," Panelist in a session on "Highly Distributed Systems," at the IEEE Symposium on Applications and the Internet, San Diego, CA, January 10, 2001.
216. "Adaptive and Reflective Middleware Systems," Panelist at the NSF Networking PI meeting, Irvine California, November 1st, 2000.
217. "Surviving the Tornado: The Best Kept Secrets of R&D Success in the Internet Age," Keck Observatory, Hawaii, October 9th, 2000.
218. "Adaptive and Reflective Middleware Systems," BBN Technologies, Boston, MA, September 27th, 2000.
219. "Distributed Application Integration: Myth or Reality? " Keynote talk at 2nd International Symposium on Distributed Objects and Applications (DOA '00), OMG, Antwerp, Belgium, September 21st, 2000.
220. "Surviving the Tornado: The Best Kept Secrets of R&D Success in the Internet Age," Keynote talk at 2nd International Symposium on Distributed Objects and Applications (DOA '00), OMG, Antwerp, Belgium, September 21st, 2000.
221. "High Confidence Adaptive and Reflective Middleware: Fact or Fiction? " Keynote talk for the IFIP Fourth International Conference on Formal Methods for Open Object-Based Distributed Systems, (FMOODS 2000), Stanford University, Stanford, CA, September 7th, 2000.
222. "Adaptive and Reflective Middleware Systems," Lockheed Martin, Ft. Worth, TX, September 6th, 2000.
223. Pattern-oriented Software Architecture: Concurrent and Networked Objects, Raytheon, San Diego, August 25, 2000.
224. "Adaptive and Reflective Middleware Systems," Rockwell/Collins, Cedar Rapids, Iowa, August 22, 2000.
225. "Adaptive and Reflective Middleware Systems," Lockheed Martin, Eagan, MN, August 21, 2000.
226. "Adaptive and Reflective Middleware Systems," Honeywell Technology Center, Minneapolis, MN, August 18, 2000.
227. "Adaptive and Reflective Middleware Systems," Raytheon, Falls Church, VA, July 12, 2000.
228. "Applying Patterns to Develop High-performance and Real-time Object Request Brokers," Lockheed Martin, Eagan, Minnesota, May 19, 2000.
229. "Patterns and Principles of Real-time Object Request Brokers," Cisco, San Jose, April 12, 2000.
230. "Patterns and Principles of Real-time Object Request Brokers," BellSouth, Atlanta, Georgia, March 3, 2000.
231. "Patterns and Principles of Real-time Object Request Brokers," Distinguished Lecturer Series, Michigan State University, East Lansing, Michigan, October 21, 1999.

232. "Towards Minimum ORBs for Wireless Devices and Networks," OPENSIG '99 Workshop, Carnegie Mellon University, Pittsburgh, October, 14-15, 1999.
233. "Applying CORBA Fault Tolerant Mechanisms to Network Management," Lucent CORBA Forum, Naperville, IL, September 28th, 1999.
234. "CORBA for Real-time and Embedded Telecom Systems," Lucent CORBA Forum, Naperville, IL, September 28th, 1999.
235. "Patterns and Principles of Real-time Object Request Brokers," BEA, Munich, Germany, September 16th, 1999.
236. "Real-time CORBA – Fact or Fiction," Siemens CORBA Day, Munich, Germany, September 15th, 1999.
237. "Patterns and Principles of Real-time Object Request Brokers," Siemens MED, Erlangen, Germany, September 13th, 1999.
238. "Patterns and Principles of Real-time Object Request Brokers," RT DII COE TWG, Boeing, Seattle, WA August 25th, 1999.
239. "Patterns for Real-time Middleware," Microsoft, Redmond, WA, August 24th, 1999.
240. "Patterns and Principles of Real-time Object Request Brokers," Lockheed Martin, Eagan, Minnesota, June 22nd, 1999.
241. "Using the ACE Framework and Patterns to Develop OO Communication Software," Dreamworks SGK, Glendale, CA, May 5th, 1999.
242. "Why Telecom Reuse has Failed and how to Make it Work for You," Keynote talk at Nortel Design Forum, Ottawa, CA, April 22nd, 1999.
243. "QoS-enabled Middleware for Monitoring and Controlling High-Speed Networks and Endsystems," Lucent Bell Labs, Murray Hill, NJ, April 15th, 1999.
244. "Optimization Patterns for High-performance, Real-time Object Request Broker Middleware," University of California, Irvine, April, 2nd, 1999.
245. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Lucent, Columbus, OH, March 18-19 and 25-26, 1999.
246. "Using Design Patterns, Frameworks, and Object-Oriented Communication Systems," Lucent, Holmdel, NJ, March 1-4, 1999.
247. Chaired a panel on "Research Directions for Middleware," NSF PI meeting, Washington, DC, January 24th, 1999.
248. "Principles and Patterns of High-performance Real-time CORBA," University of Southern California, Los Angeles, CA, December 10th, 1998.
249. "Real-time CORBA for Telecom – Fact or Fiction? ," Bellcore, Morristown, NJ, December 1st, 1998.
250. "Design Patterns for Real-time Object Request Brokers," Silicon Valley Patterns Group, San Francisco, November 15, 1998.
251. "Why Reuse has Failed and how to Make it Work for You," Keynote talk at Lucent Software Symposium, October 27th, Murray Hill, NJ, 1998.
252. "Real-time CORBA – Fact or Fiction," Lucent CORBA Forum, Holmdel, NJ, September 29, 1998.

253. "Applying Software Design Patterns and Framework to Telecommunication Applications," Nortel Advanced Software Computing and Technology, Monday, April 6, 1998, Ottawa, Canada.
254. "Patterns and Performance of Real-time Object Request Brokers," University of California, Santa Barbara, February 20, 1998.
255. "Principles and Patterns of High-performance, Real-time Object Request Brokers," University of Frankfurt, Germany, February 12th, 1998.
256. "Principles and Patterns of High-performance, Real-time Object Request Brokers," University of Illinois, Urbana-Champaign November 12th, 1997.
257. "Principles and Patterns of High-performance, Real-time Object Request Brokers," University of Missouri, Kansas City, October 31st, 1997.
258. "Principles and Patterns of High-performance, Real-time Object Request Brokers," IBM T.J. Watson Research, September 15, 1997.
259. "Principles and Patterns of High-performance, Real-time Object Request Brokers," University of California, Santa Barbara, August 21st, 1997.
260. "Principles and Patterns of High-performance, Real-time Object Request Brokers," Lucent Technologies, Naperville, IL August 19th, 1997.
261. "Mastering Software Complexity with Reusable Object-Oriented Frameworks, Components, and Design Patterns," 3rd NSA Software Reuse Symposium, August 20th, 1997.
262. "Principles and Patterns of High-performance, Real-time Object Request Brokers," University of Utah, Salt Lake City, Utah, August 11th, 1997.
263. "Using the ACE Framework and Design Patterns to Develop Object-Oriented Communication Software," CERN, Switzerland, July 18th, 1997.
264. "Principles and Patterns of High-performance, Real-time Object Request Brokers," CHOOSE symposium, Zurich, Switzerland, July 17th, 1997.
265. Invited keynote speaker for 2nd Component's User Conference, Munich Germany, July 1997.
266. "Principles and Patterns of High-performance, Real-time Object Request Brokers," Lucent Bell Laboratories, Murray Hill, New Jersey, July 9th, 1997.
267. "Using the ACE Framework and Design Patterns to Develop Object-Oriented Communication Software," Lockheed Martin Tactical Systems, Minneapolis, Minnesota, June 26th, 1997.
268. QoS for Distributed Object Computing Middleware – Fact or Fiction? , panel at the Fifth International Workshop on Quality of Service (IWQoS '97), May 22nd, 1997, Columbia University, NYC, USA.
269. "Design Patterns and Frameworks for Developing Object-oriented WWW Clients and Servers," Carleton University, April 11th, 1997.
270. "Principles and Patterns of High-performance, Real-time Object Request Brokers," University of Maryland, College Park, Maryland, April 2nd, 1997.
271. "A High-Performance End system Architecture for Real Time COBRA," SPARTAN Symposium sponsored by US Sprint, Lawrence Kansas, March 18th, 1997.
272. "Experience with CORBA for Communication Systems," Motorola, Chicago, January 24th, 1997.

273. "High-performance CORBA," Bay Area Object Interest Group, Stanford Linear Accelerator Center, California, December 5th, 1996.
274. "Gigabit CORBA – An Architecture for High-performance Distributed Object Computing," Numerical Aerodynamic Simulation group, NASA, Moffett Field, California, December 3rd, 1996.
275. "Towards High-performance, Real-time CORBA," Distinguished Lecturer at Kansas State University, Manhattan, Kansas, November 7th, 1996.
276. "Gigabit CORBA – An Architecture for High-performance Distributed Object Computing," University of California, Los Angeles, October 3rd, 1996.
277. "Design Patterns and Frameworks for Object-Oriented Communication Software," NSA Software Reuse Symposium, August 28th, 1996.
278. "CORBA – the Good, the Bad, and the Ugly," Lucent Bell-Labs, Naperville, IL, August 22nd, 1996.
279. "Components: the Good, the Bad, and the Ugly," keynote talk for the 1st Components Users Conference, SIEMENS, Munich, Germany, July 15th, 1996.
280. "Design Patterns for Object-Oriented Communication Software," IONA Technologies, Ltd, Dublin, Ireland, July 12th, 1996.
281. "OO Design Patterns and Frameworks for Communication Software," Siemens Corporate Research, Princeton, New Jersey, June 27, 1996.
282. "OO Design Patterns for Concurrent, Parallel, and Distributed Systems," IBM Centre for Advanced Studies, North York, Ontario, Canada, June 17, 1996.
283. "Distributed Object Computing with CORBA", Bell Laboratories, Murray Hill, New Jersey, June 11-12th, 1996.
284. "Design Patterns for Object-Oriented Communication Software," Carleton University, Ottawa, Canada, May 21st, 1996.
285. "Integrating LAN-WAN-Celestial Networks with Design Patterns," Featured technical session at the Object World East conference, Boston, MA, May 9th, 1996.
286. "Using Design Patterns to Develop Object-Oriented Communication Software Frameworks and Applications," McMaster's University, Hamilton, Canada, May 2nd, 1996.
287. "Towards Gigabit CORBA – A High-Performance Architecture for Distributed Object Computing," University of Nevada, Reno, April 25th, 1996.
288. "Domain Analysis: From Tar Pit Extraction to Object Mania? " Panelist at the 4th International Conference on Software Reuse, Orlando, Florida, April 25th, 1996. (other panelists include Spencer Peterson, SEI CMU, Mark Simos, Organon Motives Inc., Will Tracz, Loral, and Nathan Zalman, BNR Inc).
289. "Concurrent Object-Oriented Network Programming with C++," Kodak Imaging Technology Center, April 19th, 1996.
290. "Using OO Design Patterns and Frameworks to Develop Object-Oriented Communication Systems," INRS/NorTel Workshop on Telecommunication Software, Montreal, CA, March 14th, 1996.

291. "Concurrent Object-Oriented Network Programming with ACE and C++," for Siemens Medical Engineering, Erlangen Germany, February 15*th*, 1996.
292. "OO Componentware" Panelist at the *OOP '96 Conference*, SIGS, Munich, Germany, February 13*st*, 1996. (other panelists included Michael Stal (Siemens AG) and Frank Buschmann (Siemens AG).
293. "Using Design Patterns to Develop High-performance Object-Oriented Communication Software Frameworks," for the Department of Information Systems, Institute of Computer Science, Johannes Kepler University of Linz, Austria, February 12*th*, 1996.
294. "The Performance of Object-Oriented Components for High-speed Network Programming," for the Digital Libraries research group at Stanford University, Palo Alto California, February 2*nd*, 1996.
295. "Distributed Object Computing with CORBA, ACE, and C++," for South Western Bell Telephone advanced distributed systems group, St. Louis, MO., January 26*th*, 1996.
296. "OO Design Patterns for Large-Scale Object-Oriented Communication Software Systems," AG Communication Systems, Phoenix, Arizona, December 11–13*th*, 1995.
297. "Experience Using OO Design Patterns to Develop Large-Scale Object-Oriented Communication Software Systems," Bell Northern Research, 7th Annual Design Forum, Ottawa, Canada, December 6*th*, 1995.
298. "Using OO Design Patterns to Develop Large-Scale Distributed Systems," Object Technology International, Ottawa, Canada, November 22*nd*, 1995.
299. "Design Patterns for Concurrent, Parallel, and Distributed Systems," North Dallas Society for Object Technology, September 13*th*, 1995.
300. "Using Design Patterns for Iridium Communication Services," at Motorola Iridium, Chandler, AZ, June 30*th*, 1995.
301. "Object Technology and the World-Wide Information Infrastructure," Panelist at ECOOP '95, Aarhus, Denmark, August 9*th*, 1995.
302. "Measuring the Performance of CORBA over ATM Networks," HP Labs, Palo Alto, CA, June 28*th*, 1995.
303. "Measuring the Performance of Object-Oriented Components for High-speed Network Programming," The C++ and C SIG user group, New York, New York, June 5*th*, 1995.
304. "An Overview of Design Patterns for Object-Oriented Network Programming," St. Louis Chapter of the ACM, St. Louis, MO, March 13*th* 1995.
305. "Design Patterns for Concurrent Object-Oriented Network Programming," Distributed Systems group at Siemens Corporate Research Center, Munich, Germany, March 3*rd*, 1995.

306. "Patterns: 'Eureka,' 'Deja-Vu,' or 'Just Say No'? " Panelist at the *OOP '95 Conference*, SIGS, Munich, Germany January 31st, 1995. (other panelists included Richard Helm, (DMR), Frank Buschmann (Siemens AG), and Dave Thomas (OTI).
307. "Developing Distributed Applications with the ADAPTIVE Communication Environment," *The 12th Annual Sun Users Group Conference*, SUG, San Francisco, California, June 17th, 1994.
308. "Flexible Configuration of High-performance Distributed Communication Systems," presented at the ETH-Zentrum in the Swiss Federal Institute of Technology, Zurich, Switzerland, May 31st, 1994.
309. "Object Oriented Techniques for Developing Distributed Applications," *Computer Science Department Colloquia*, California State University Northridge, December 7th, 1993.
310. "Hosting the ADAPTIVE System in the *x*-Kernel and System V STREAMS," *The x-Kernel Workshop*, IEEE, Tucson, Arizona, November 10th, 1992.
311. "An Environment for Controlled Experimentation on the Performance Effects of Alternative Transport System Designs and Implementations," IBM T. J. Watson Research Center, Hawthorne, New York, September 10th, 1992.

Colloquia, Seminars, and Tutorials

1. "The C++ Standard Template Library," Qualcomm, San Diego, February 16-19, 2016.
2. "The C++ Standard Template Library," Qualcomm, San Diego, October 13-16, 2015.
3. "Pattern-Oriented Java Concurrency," InformIT Webinar, May 14th, 2015.
4. "Pattern-Oriented Concurrent Programming with Java," OOP Conference, Munich, Germany, January 30th, 2015.
5. "Concurrent Programming in Android," OOP Conference, Munich, Germany, January 29th, 2015.
6. "The C++ Standard Template Library," Qualcomm, San Diego, October 14-17, 2014.
7. "The C++ Standard Template Library," Qualcomm, San Diego, August 5-8, 2014.
8. "Pattern-Oriented Software Architecture for Concurrent and Networked Software," July 28-31, 2014.
9. "The C++ Standard Template Library," Qualcomm, San Diego, August 5-8, 2014.
10. "The C++ Standard Template Library," Qualcomm, India, March, 2014.
11. "The C++ Standard Template Library," Qualcomm, San Diego, CA, January 23-34, 2014.
12. "The C++ Standard Template Library," Qualcomm, San Diego, CA, October 16-17th, 2013.

13. "Patterns and Frameworks for Concurrent and Networked Software," 2013 International Summer School on Trends in Computing Tarragona, Spain, July 25-26, 2013.
14. "The C++ Standard Template Library," Qualcomm, San Diego, CA, January 23-24th, 2013.
15. "The C++ Standard Template Library," Qualcomm, San Diego, CA, October 4-5th, 2012.
16. "Embedded Systems Patterns for C Developers," Qualcomm, San Diego, CA, August 28th, September 11th, September 25th, October 9th, October 23rd, and November 6th, 2012.
17. "Embedded Systems Patterns for C Developers," Qualcomm, San Diego, CA, August 14-15th, 2012.
18. "The C++ Standard Template Library," Qualcomm, San Diego, CA, May 15-18th, 2012.
19. "The C++ Standard Template Library," Qualcomm, San Diego, CA, January 25-26th, 2012.
20. "Object-Oriented Software Patterns and Frameworks," Qualcomm, San Diego, CA, October 11-12th, 2011.
21. "The C++ Standard Template Library," Qualcomm, San Diego, CA, May 11-12th, 2011.
22. "The C++ Standard Template Library," Qualcomm, San Diego, CA, January 25-26, 2011.
23. "Pattern-Oriented Software Architecture: A Pattern Language for Concurrent and Networked Software," SPLASH 2010, October 17-21, 2010, Reno, Nevada.
24. "Pattern-Oriented Software Architectures - Patterns and Frameworks for Concurrent and Networked Software," ProObject, Hanover, MD, August 11th, 2010.
25. "Pattern-Oriented Software Architecture: Patterns for Concurrent and Networked Embedded Systems," Qualcomm, Bangalore, India, June 21-22, 2010
26. "Pattern-Oriented Software Architecture: Patterns for Concurrent and Networked Embedded Systems," Qualcomm, Hyderabad, India, June 24-25, 2010.
27. "Pattern-Oriented Software Architecture: A Pattern Language for High Quality and Affordable Distributed Computing Systems," IEEE Webinar Series, June 10th, 2010.
28. "The C++ Standard Template Library," Qualcomm, San Diego, CA, May 12-13, 2010.
29. "Design Patterns for Understanding Middleware and Component Infrastructures," 6th USENIX Conference on Object-Oriented Technologies and Systems, January 29, 2001, San Antonio, TX.
30. "The C++ Standard Template Library," Qualcomm, San Diego, CA, December 16-17, 2009.
31. "Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing," OOPSLA 2009, Orlando, FL, October, 2009.

32. "The C++ Standard Template Library," Qualcomm, San Diego, CA, September 15-16, 2009.
33. "Networked Embedded Systems Patterns for C Developers," Qualcomm, San Diego, CA, June 11-12, 2009.
34. "Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing," Software Architecture Technology Users' Network (SATURN) workshop May 5, 2009 in Pittsburgh, PA.
35. "The C++ Standard Template Library," Qualcomm, San Diego, CA, January 29-30, 2009.
36. "Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing," IEEE Webinar Series, January 8th, 2009.
37. "Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing," OOPSLA 2008, Nashville, TN, October 20, 2008.
38. "The Data Distribution Service for Real-time Systems," OOPSLA 2008, Nashville, TN, October 19, 2008.
39. "Object-Oriented Patterns for Concurrent and Networked Applications," Qualcomm, San Diego, CA, August 5-6th, 2008.
40. "The C++ Standard Template Library," Qualcomm, San Diego, NJ, July 29-30, 2008.
41. "Object-Oriented Patterns and Frameworks with C++," Qualcomm, San Diego, CA, June 12-13, 2008.
42. "The C++ Standard Template Library," Qualcomm, New Jersey, May 5-6, 2008.
43. "Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing," Software Architecture Technology Users' Network (SATURN) workshop April 28 - May 1, 2008 in Pittsburgh, PA.
44. Developing Distributed Computing Systems with Patterns and Middleware, UCLA Extension, February 19-21, 2008.
45. Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing, OOPSLA 2007, Montreal, CA, October 24, 2007.
46. Object-Oriented Design and Programming with Patterns, Frameworks, and Middleware, Qualcomm, New Jersey, September 27-28, 2007.
47. Object-Oriented Design and Programming with Patterns, Frameworks, and Middleware, Qualcomm, San Diego, CA, August 21-22, 2007.
48. Lightweight CORBA Component Model, 8th OMG Real-time/Embedded CORBA workshop, Washington DC, July 9–12, 2007.
49. Model-Driven Engineering for Distributed Real-time and Embedded Systems, 13th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2007), Bellevue, WA, United States April 3-6, 2007.
50. "Improving Product Reliability and ROI Through Effective Software Reuse," Qualcomm, San Diego, CA, March 27th, 2007.
51. "Developing Distributed Computing Systems with Patterns and Middleware," UCLA Extension, February 21-23, 2007.
52. "POSA: Patterns for Concurrent and Distributed Systems," OOP, Munich, Germany, January 22, 2007.

53. "Meeting the Challenges of Software-Intensive Embedded Systems," OOP, Munich, Germany, January 23, 2007.
54. "Object-Oriented Design and Programming with Patterns, Frameworks, and Middleware," Qualcomm, San Diego, CA, January 10-11, 2007.
55. "Model-Driven Development of Distributed Systems," OOPSLA 2006, Portland, OR, October 22-26, 2006.
56. "Pattern-Oriented Software Architecture: Patterns for Concurrent and Networked Objects," OOPSLA 2006, Portland, OR, October 22-26, 2006.
57. "Model-Driven Engineering of Distributed Systems," MODELS 2006, Genova, Italy, October 1, 2006.
58. "Distributed Real-time and Embedded Systems," Advanced Institute of Information Technology, Seoul, Korea, August 7-11 2006.
59. "Lightweight CORBA Component Model," 7th OMG Real-time/Embedded CORBA workshop, Washington DC, July 10–13, 2006.
60. "How to Use ACE Effectively," Trion World Network, Austin, TX, June 19-21, 2006.
61. "Improving Product Reliability and ROI Through Effective Software Reuse," Qualcomm, San Diego, CA, June 15, 2006.
62. "Object-Oriented Design and Programming with Patterns, Frameworks, and Middleware," Qualcomm, San Diego, CA, June 13-14, 2006.
63. "Object-Oriented Design and Programming with Patterns, Frameworks, and Middleware," Qualcomm, San Diego, CA, Feb 9-10, 2006.
64. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems, University of California, Los Angeles Extension, January 18-20st, 2006."
65. "Model Driven Development of Distributed Real-time and Embedded Systems," at the OOP conference, January 17, 2006, Munich, Germany.
66. "Pattern-Oriented Software Architecture," at the OOP conference, January 16, 2006, Munich, Germany.
67. "Model Driven Development: State of the Art," at the OOP conference, January 16, 2006, Munich, Germany.
68. "Concurrent C++ Network Programming with Patterns and Frameworks," C++ Connections: 20 Years of C++ conference, November 11, 2005, Mandalay Bay, Las Vegas, NV.
69. "Pattern-Oriented Software Architecture: Patterns for Concurrent and Distributed Systems," OOPSLA 2005, San Diego, October 17th, 2005.
70. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," BAE Systems, Greenlawn, New York, August 25, September 2-3.
71. "Lightweight CORBA Component Model," 6th OMG Real-time/Embedded CORBA workshop, Washington DC, July 11–14, 2005.
72. "Model Driven Development for Distributed Real-time and Embedded Systems," OMG Information Days: MDA - Frankfurt, Germany, June 9th, 2005

73. "Model Driven Development for Distributed Real-time and Embedded Systems," OMG Information Days: MDA - Munich, Germany, June 7th, 2005.
74. "Model Driven Development for Distributed Real-time and Embedded Systems," OMG Information Days: MDA - Zurich, Switzerland, June 1st, 2005.
75. Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," BAE Systems, Wayne, New Jersey, May 13, 16, 19, 23, 27, 2005.
76. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," BAE Systems, Wayne, New Jersey, February 18th, February 22nd, March 1, 8, and 15 2005.
77. "Pattern-Oriented Software Architectures for Distributed Systems" the OOP conference, January 28, 2005, Munich, Germany.
78. "Research on Model Driven Development of Distributed Real-time and Embedded Systems," the OOP conference, January 26, 2005, Munich, Germany.
79. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Los Angeles Extension, January 19-21st, 2005.
80. Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems, BAE Systems, Wayne, New Jersey, October 29, November 1, 8, 15, 22, 2004.
81. "Pattern-Oriented Software Architectures for Distributed Systems," OOPSLA 2004, Vancouver, British Columbia, October 25th, 2004.
82. "Notes on the Forgotten Craft of Software Architecture", OOPSLA 2004, Vancouver, British Columbia, October 25th, 2004.
83. "Model Driven Architecture with QoS-enabled component middleware," MDE for Embedded Systems, Brest, France, September 10th 2004.
84. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Qualcomm, San Diego, CA, Jan 7-6, 2005.
85. "Object-Oriented Design and Programming with Patterns, Frameworks, and Middleware," Qualcomm, San Diego, CA, Jan 9-10, 2005.
86. "Using the Lightweight CORBA Component Model to Develop Distributed Real-time and Embedded Applications," OMG Workshop on Distributed Object Computing for Real-time and Embedded Systems, July 12th, 2004, Reston, VA.
87. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Los Angeles Extension, July 7-9th, 2004.
88. "Model Driven Middleware for Distributed Real-time and Embedded Systems," Keynote talk for the SIGS Software Engineering Today conference in Zurich, Switzerland, May 4-5, 2004.
89. Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems, University of California, Los Angeles Extension, January 21st-23rd, 2004.
90. Patterns and Frameworks for Concurrent Distributed Systems, SIGS OOP Conference, Munich, Germany, January 19th, 2004.

91. Middleware for Distributed Real-time and Embedded Systems, SIGS OOP Conference, Munich, Germany, January 19th, 2004.
92. "Pattern-Oriented Software Architectures for Networked and Concurrent Applications," OOPSLA 2003, Anaheim, CA, October 27, 2003.
93. The JAOO 2003 conference, September 22-26, Aarhus, Denmark.
94. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Los Angeles Extension, July 9-11th, 2003.
95. "Patterns, Frameworks, and Middleware: Their Synergistic Relationship," Frontiers of Software Practice, International Conference on Software Engineering, Portland, Oregon, May 7, 2003.
96. "Pattern-Oriented Distributed Systems Architecture," International Conference on Software Engineering, Portland, Oregon, May 5, 2003.
97. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Los Angeles Extension, January 22nd-24th, 2003.
98. "Patterns and Application Experiences for Real-time Object Request Brokers," OOPSLA 2002, Seattle, Washington, November, 2002.
99. "Pattern-Oriented Software Architectures for Networked and Concurrent Applications," OOPSLA 2002, Seattle, Washington, November, 2002.
100. Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems, Raytheon, St. Petersburg, FL, September 3–5, 2003.
101. Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems, University of California, Los Angeles Extension, July 22nd-24th, 2002.
102. "Policies and Patterns for High-performance, Real-time Object Request Brokers," Mercury Computer Systems, Tysons Corner, VA, November Feb 7, 2002.
103. Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems, University of California, Los Angeles Extension, January 23rd-25th, 2002.
104. "Policies and Patterns for High-performance, Real-time Object Request Brokers," Raytheon, Rosslyn, VA, November 12th, 2001.
105. "Pattern-Oriented Software Architecture: Patterns for Concurrent and Networked Objects," OOPSLA 2001, October 15th, 2000, Minneapolis, Minnesota.
106. "Policies and Patterns for High-performance, Real-time Object Request Brokers," International Symposium on Distributed Object Applications (DOA), Rome, September 17-20, 2001.
107. "Policies and Patterns for QoS-enabled Middleware," The JAOO 2001 conference, September 10-14, Aarhus, Denmark.
108. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Los Angeles Extension, July 23rd-25th, 2001.

109. "Policies and Patterns for High-performance, Real-time Object Request Brokers," OMG Second Workshop on Real-time and Embedded Distributed Object Computing on June 4-7, 2001 in Herndon, VA, USA.
110. "Principles and Patterns of High-performance, Real-time Object Request Brokers," OOP conference, Munich, Germany, January 23, 2001.
111. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Los Angeles Extension, January 3-5, 2001.
112. "Patterns for Concurrent and Distributed Objects," OOPSLA 2000, October 16th, 2000, Minneapolis, Minnesota.
113. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Berkeley Extension, May 24-26, 2000.
114. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Jet Propulsion Laboratory, Pasadena, CA, April, 2000.
115. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Los Angeles Extension, March 27-31, 2000.
116. "Optimizing Middleware to Support High-Performance Real-time Distributed and Embedded Systems," OOP conference, Munich, Germany, January 27, 2000.
117. "Effective Architectures for DOC," OOP conference, Munich, Germany, January 24, 2000.
118. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California, Berkeley Extension, December 13-15, 1999.
119. "Middleware Techniques and Optimizations for Real-time Embedded Systems," 12th International Symposium On System Synthesis, IEEE, San Jose, CA, USA November, 11, 1999
120. "Patterns and Principles of Real-time Object Request Brokers," OOPSLA 1999, ACM, Denver, Colorado, November 1-5, 1999.
121. "Using Design Patterns, Frameworks and CORBA to Reduce the Complexity of Developing Reusable Large-Scale Object-Oriented Concurrent Communication Components and Systems," Fifth IEEE International Conference on Engineering of Complex Computer Systems, Las Vegas, Nevada, October 18-21, 1999
122. "Distributed Technologies," Motorola, Schaumburg, IL, August 10-12, 1999.
123. "Patterns and Principles of Real-time Object Request Brokers," the 3rd Components Users Conference, SIEMENS, Munich, Germany, July 12th, 1999.
124. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Lucent, Naperville, IL, June 23-24 and June 30 - July 1st, 1999.
125. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Motorola Software Symposium, Ft. Lauderdale, Florida, June 21st, 1999.

126. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California Los Angeles Extension, June 2-4, 1999.
127. "Concurrent Object-Oriented Network Programming and Distributed Object Computing," University of California Berkeley Extension, May 19-21, 1999.
128. "Patterns and Principles of Real-time Object Request Brokers," 5th USENIX Conference on Object-Oriented Technologies and Systems, May 4, 1999, San Diego, CA.
129. "Real-time CORBA for Telecom – Fact or Fiction? " Nortel Design Forum, Ottawa, CA, April 22, 1999.
130. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Lucent, Columbus, OH, March 18-19 and 25-26, 1999.
131. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Lucent, Holmdel, NJ, March 1-4, 1999.
132. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Lucent/Octel, Milpitas, CA, December 14-16, 1998.
133. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California Los Angeles Extension, December 8-10, 1998.
134. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Motorola, Schaumburg, IL, December 2-4, 1998.
135. "Concurrent Object-Oriented Network Programming and Distributed Object Computing," University of California Berkeley Extension, November 16-18, 1998.
136. "Using Design Patterns and Frameworks to Develop Object-Oriented Communication Software," OOPSLA 1998, October 19th, 1998, Vancouver, British Columbia.
137. "High-Performance CORBA," Lucent CORBA Forum, Holmdel, NJ, September 29, 1998.
138. "Writing Efficient Multi-Thread CORBA Applications," the 3rd Components Users Conference, SIEMENS, Munich, Germany, July 10, 1998.
139. "Using Design Patterns and Frameworks to Develop Object-Oriented Communication Software," UCLA extension course, Milan, Italy, June 29 - July 1, 1998.
140. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," Lucent, Naperville, IL, June 8-11, 1998.
141. "Patterns and Performance of Real-time Object Request Brokers," Fourth IEEE Real-Time Technology and Applications Symposium (RTAS), Denver, Colorado, June 5, 1998.
142. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California Los Angeles Extension, June 1-3, 1998.

143. "Patterns and Principles of Real-time Object Request Brokers," NSA, Ft. Meade, MD, March 22, 1998.
144. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems, Crosskeys, Ottawa Canada, March 19-21, 1998.
145. "Concurrent Object-Oriented Network Programming and Distributed Object Computing," University of California Berkeley Extension, March 4-6, 1998.
146. "Building Distributed Communication Software with CORBA," the Motorola Systems Symposium, February, 1998, Austin, Texas, USA.
147. "Introduction to Distributed Objects with CORBA," SIGS OOP '98, February 9-13, 1998, Munich, Germany.
148. "Design Patterns for Developing and Using CORBA Object Request Brokers," SIGS OOP '98, February 9-13, 1998, Munich, Germany.
149. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems, Lucent Technologies, Whippany, NJ, January 5-6, 1998.
150. "Using Design Patterns, Frameworks, and CORBA to Develop Object-Oriented Communication Systems," University of California Los Angeles Extension, December 10-12, 1997.
151. "Concurrent Object-Oriented Network Programming and Distributed Object Computing," University of California Berkeley Extension, December 10-12, 1997.
152. "Using Design Patterns and Frameworks to Develop Object-Oriented Communication Systems," Motorola Cellular Infrastructure Group, Arlington Heights, Illinois, December 1 - 3, 1997.
153. "Using Design Patterns and Frameworks to Develop Object-Oriented Communication Systems," TOOLS Pacific '97, Melbourne, Australia November 24 - 27, 1997.
154. "Using Design Patterns and Frameworks to Develop Object-Oriented Communication Systems" for the IEEE GLOBECOM '97 conference, Phoenix, AZ, November 4-8, 1997.
155. "High-performance Distributed Object Computing with CORBA," IEEE International Conference on Network Protocols, Atlanta, GA, October 28th, 1997.
156. "Using Design Patterns and Frameworks to Develop Object-Oriented Communication Systems," OOPSLA 1997, ACM, Atlanta, GA, October 6-7th, 1997.
157. "Using Design Patterns and Frameworks to Develop Object-oriented Communication Systems," 24th International Conference on Technology of Object-Oriented Languages and Systems (TOOLS Asia '97). Beijing, China, September 22, 1997.
158. "Principles and Patterns of Distributed Object Computing Systems," for the ACM Principles of Distributed Computing Conference (PODC), Santa Barbara, CA, August 21st, 1997.

159. "Distributed Object Computing with CORBA and ACE," Alta Software, Jacksonville, FL, June 4-5th, 1997.
160. "Distributed Object Computing with CORBA", Object Expo, NY, NY, June 2nd, 1997.
161. "Concurrent Object-Oriented Network Programming and Distributed Object Computing," University of California Berkeley Extension, May 28-30, 1997.
162. "Patterns and Principles of Real-time Object Request Brokers," National Security Agency, Ft. Meade, MD, May 13th, 1997.
163. "Building Distributed Communication Software with CORBA," the Motorola Systems Symposium, March, 1997, Chandler, AZ, USA.
164. "Evaluating Concurrency Models for CORBA Servers," the 2nd Components Users Conference, SIEMENS, Munich, Germany, July 14th, 1997.
165. "Design Patterns for Evolving System Software Components from UNIX to Windows NT," the 2st Components Users Conference, SIEMENS, Munich, Germany, July 14th, 1997.
166. "Techniques and Patterns for Distributed Object Computing with CORBA and C++," University of California Berkeley Extension, December 4-6, 1996.
167. "Design Patterns for Concurrent Object-Oriented Programming with ACE and C++," C++ World, Dallas, TX, November 11th, 1996.
168. "Implementing Concurrent CORBA Applications with Multi-Threaded Orbix and ACE," C++ World, Dallas, TX, November 12th, 1996.
169. "Why Reuse has Failed, and How You Can Make it Work for You," Berne Technology Forum 1996, Berne, Switzerland, October 18, 1996.
170. "Introduction to Distributed Object Programming with CORBA," the Local Computer Networks '96 conference, IEEE, Minneapolis, Minnesota, October 13, 1996.
171. "Object-Oriented Design Patterns for Concurrent, Parallel, and Distributed Systems," the OOPSLA 1996 conference, ACM, San Jose, California, October, 1996.
172. "OO Design Patterns Network Programming in C++," Object Expo Europe, London, England, September 23rd, 1996.
173. "Effective Multithreaded CORBA Programming," Object Expo Europe, London, England, September 24th, 1996.
174. "Workshop on Object Oriented Technologies," Mitsubishi, July 22nd to July 26th, 1996, Kobe, Japan.
175. "Evaluating Concurrency Models for CORBA Servers," the 1st Components Users Conference, SIEMENS, Munich, Germany, July 15th, 1996.
176. "Design Patterns for Evolving System Software Components from UNIX to Windows NT," the 1st Components Users Conference, SIEMENS, Munich, Germany, July 15th, 1996.
177. "OO Design Patterns for Concurrent, Parallel, and Distributed Systems," the *2nd Conference on Object-Oriented Technology*, USENIX, Toronto, Canada, June 17, 1996.

178. "OO Design Patterns for Concurrent, Parallel, and Distributed Systems," the *3rd Conference on Object-Oriented Technology*, USENIX, Portland, Oregon, June 16th, 1996.
179. "OO Design Patterns for Network Programming in C++," the *Object Expo '96 Conference*, SIGS, Sydney, Australia, June 3rd, 1996.
180. "Effective Multi-threaded CORBA Programming Programming," the *Object Expo '96 Conference*, SIGS, Sydney, Australia, June 5th, 1996.
181. "Concurrent Object-oriented Network Programming with C++," University Of California Berkeley Extension, Berkeley, California, May 22nd––24th, 1996.
182. "Experience Developing Reusable Software Using Object-Oriented Design Patterns and Frameworks," the *4th International Conference on Software Reuse*, Orlando, Florida, USA April 23-26, 1996.
183. "Techniques for Object-Oriented Network Programming," the *OOP Conference*, SIGS, Munich, Germany, Feb 14th, 1996.
184. "Using Object-Oriented Design Patterns to Develop Large-Scale Distributed Systems," the *OOP Conference*, SIGS, Munich, Germany, Feb 13th, 1996.
185. "Concurrent Object-oriented Network Programming with C++," University Of California Berkeley Extension, Berkeley, California, November 30th-December 1st, 1995.
186. "Using Object-Oriented Design Patterns to Develop Large-Scale Distributed Systems," the *4th C++ World Conference*, SIGS, Chicago, Illinois, October 31st, 1995.
187. "Techniques for Object-Oriented Network Programming," the *4th C++ World Conference*, SIGS, Chicago, Illinois, October 31st, 1995.
188. "Experience using OO Design Patterns to Develop Large-scale Distributed Communication Systems," *OOPSLA 1995 Conference* in Austin, Texas, October 1995.
189. "Concurrent Object-oriented Network Programming with C++," the *9th European Conference on Object-Oriented Programming (ECOOP)*, Aarhus, Denmark, August, 1995.
190. "Concurrent Object-Oriented Network Programming with C++," the *1st Conference on Object-Oriented Technology*, USENIX, Monterey, California, June 23, 1995.
191. "Design Patterns for Concurrent and Distributed Systems," the *Object Expo '95 Conference*, SIGS, New York, NY, June 5th 1995.
192. "Object Oriented Network Programming," the *Object Expo '95 Conference*, SIGS, New York, NY, June 5th, 1995.
193. "Software Construction with Active Objects in C++," the *OOP '95 Conference*, SIGS, Munich, Germany January 31, 1995.

194. "Object-Oriented Concurrent Programming with C++," the *OOP '95 Conference*, SIGS, Munich, Germany January 31, 1995.
195. "Concurrent Object-Oriented Programming," the *Winter USENIX Conference*, USENIX, New Orleans, Louisiana, January, 1995.
196. "Object-Oriented Network Programming with C++," the *3rd C++ World Conference*, SIGS, Austin, Texas, November 14, 1994.
197. "Object-Oriented Techniques for Dynamically Configuring Concurrent Distributed Applications," the *9th OOPSLA 1994*, ACM, Portland, Oregon, October 23, 1994.
198. "Object-Oriented Network Programming," the *6th C++ Conference*, USENIX, Cambridge, Massachusetts, April 11, 1994.
199. "Object-Oriented Techniques for Developing Extensible Network Servers," the *2nd C++ World Conference*, SIGS, Dallas, Texas, October 19, 1993.

Professional Activities

Editorial Activities

1. Guest co-editor for a special issue of the Springer Journal Annals of Telecommunications on "Middleware for Internet distribution in the context of Cloud Computing and the Internet of Things," 2016, with Gordon Blair and Chantal Taconet.
2. Guest co-editor of the Proceedings of the IEEE special issue on Applications of Augmented Reality Environments.
3. Guest co-editor of the International Journal of Network Protocols and Algorithms (NPA) Special Issue on Data Dissemination for Large scale Complex Critical Infrastructures, 2010.
4. Wrote the foreword to the book *Patterns of Parallel Software Design* by Jorge Luis Ortega Arjona, Wiley, 2010.
5. Editorial board member of the Springer Journal of Internet Services and Applications (JISA).
6. Editorial board member of the Transactions on Pattern Languages of Programming (TPLoP) published by Springer-Verlag.
7. Wrote the foreword to the book *Practical Software Factories in .NET*, by Gunther Lenz and Christoph Wienands, Apress, 2006.
8. Guest editor of the IEEE Computer Special Issue on Model Driven Development, February 2006.
9. Guest co-editor of IEEE Network special issue on "Middleware Technologies for Future Communication Networks," February 2004 (co-editors with Gordon Blair and Andrew Campbell).
10. Editorial board member of the Springer Journal of Aspect-Oriented Software Development.

11. Wrote the foreword to the book *Fundamentals of Distributed Object Systems: The CORBA Perspective*, by Zahir Tari and Omran Bukhres, Wiley and Sons, 2001.
12. Wrote the foreword to the book *Design Patterns in Communication Software*, edited by Linda Rising, Cambridge University Press, 2000.
13. Guest editor of the Special Issue on Components and Patterns for *The Journal of Theory and Practice of Object Systems*, Wiley & Sons, to appear 2002.
14. Invited editorial on "Trends in Distributed Object Computing" for the special issue on Distributed Object-Oriented Systems appearing in the Parallel and Distributed Computing Practices journal, edited by Maria Cobb and Kevine Shaw, Vol. 3, No. 1, March 2000.
15. Co-editor of "Building Application Frameworks: Object-Oriented Foundations of Framework Design," John Wiley & Sons, 1999 (co-editors are Mohamed Fayad and Ralph Johnson), ISBN 0-471-24875-4.
16. Co-editor of "Implementing Application Frameworks: Object-Oriented Frameworks at Work," John Wiley & Sons, 1999 (co-editors are Mohamed Fayad and Ralph Johnson), ISBN 0-471-25201-8.
17. Guest editor of the Special Issue on OO Application Frameworks for the Communications of the ACM, (co-editor Mohamed Fayad), ACM, October, 1997.
18. Guest editor of the special issue on Distributed Object Computing for USENIX Computing Systems Journal, November/December, 1996.
19. Guest editor of a feature topic on Distributed Object Computing for IEEE Communications Magazine, February, 1997.
20. Wrote the foreward for Dr. Nayeem Islam's book on *Distributed Objects: Methodologies for Customizing Operating Systems* (IEEE Computer Society Press, 1996).
21. Guest editor of the Special Issue on Patterns and Pattern Languages for Communications of the ACM, (co-editors Ralph Johnson and Mohamed Fayad), ACM, October, 1996.
22. Co-editor of a book entitled "Pattern Languages of Program Design," Addison-Wesley, 1995 (co-editor is Jim Coplien, Bell Labs).
23. Editor of the Patterns++ section of the C++ Report Magazine, April 1997 - March 1998.
24. Editor-in-chief of the C++ Report Magazine, January 1996 - February 1997.
25. Editorial board member of the IEEE Computer Society - Computer Science & Engineering Practice Board.

Program Chairmanships and Conference Organization

1. General Chair of the Software Product Line Conference, Nashville TN, Jule/August, 2015.
2. Program Chair of the Interoperable Open Architecture 2013 conference, September 10-11, 2013, Washington, DC.
3. Program Chair of the NSF Workshop on Computing Clouds for Cyber-Physical Systems, March 15th, 2013, Ballston, VA.

4. Program Chair of the Interoperable Open Architecture 2012 conference, October 29-31, 2012, London, UK.
5. Program co-chair for the 1st International Symposium on Secure Virtual Infrastructures (DOA-SVI'11), 17-19 Oct 2011, Crete, Greece.
6. Program co-chair for the COMmunication System softWAre and middleware (Comsware) conference, Helsinki, Finland, August 2010.
7. Doctoral symposium chair for OOPSLA 2009, Orlando Floria, October 25-29, 2009.
8. General co-chair for the 3rd ACM International Conference on Distributed Event-Based Systems (DEBS 2009), July 6-9, 2009 - Nashville, TN, USA.
9. Member of the ISORC 2009 advisory and publicity committee for ISORC 2009, March 17-20, 2009, Toyko, Japan.
10. Area Coordinator for the Integrating Systems of Systems using Services topic at the 6th International Conference on Service Oriented Computing, Sydney (Australia), December 1st - 5th, 2008.
11. Member of the Advisory and Publicity Committee for ISORC 2008, Orlando, Florida, May 5 -7, 2008.
12. Co-chair of the Middleware for Network Eccentric and Mobile Applications (MiNEMA.08) Workshop co-located with ACM EuroSys Conference, March 31 - April 1, 2008, Glasgow, Scotland.
13. General chair of the ACM/IEEE 10th International Conference on Model Driven Engineering Languages and Systems (MODELS 2007), Nashville TN, September 30-October 5, 2007.
14. Area co-coordinator for the Quality of Service research track at the The Fifth International Conference on Service-Oriented Computing, September 17-20, 2007, Vienna, Austria.
15. Program co-chair of the NSF workshop on New Research Directions in Composition and Systems Technology for High Confidence Cyber Physical Systems, July 9, 2007.
16. Program co-chair for the Science of Design Principal Investigators workshop, February 28 to March 2, 2007.
17. Program co-coordinator for SOA Runtime area of the 4th International Conference on Service Oriented Computing Chicago, USA, December 4-7, 2006.
18. Program co-chair of the NSF/NCO Workshop on High-Confidence Software Platforms for Cyber-Physical Systems (HCSP-CPS) Workshop systems, November 30th to December 1st, 2006, Alexandria, VA.
19. Panels chair for the MoDELS 2006 conference, Genova Italy, Oct. 2-6, 2006.
20. Program Co-Chair of the Generative Programming and Component Engineering (GPCE) Conference, Portland, OR, October 2006 (collocated with OOPSLA '06).
21. Program Chair of the NSF/NCO Workshop on New Research Directions in High Confidence Software Infrastructure for Distributed Real-time and Embedded (DRE) systems, July 10th, 2006, Fairfax VA.

22. Program Co-Chair of the NSF/NCO High Confidence Medical Device Software and Systems (HCMDSS) Workshop, May 2005, University of Pennsylvania, Philadelphia, Pennsylvania.
23. Track Vice Chair for Real-time Middleware and Software Engineering for the Real-time Systems Symposium, Lisbon, Portugal, December, 2004.
24. Program Co-chair for the NSF/NCO Planning Meeting for the High Confidence Medical Device Software and Systems (HCMDSS) Workshop, November 16-17, 2004, Arlington, VA.
25. Program chair for 19th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOSPLA), October 24-28, 2004, Vancouver, British Columbia, Canada.
26. General co-chair of the IEEE Real-Time and Embedded Technology and Applications Symposium, May 25 – 28, 2004, Toronto, Canada.
27. Program chair of the CCM Workshop, December 10th, 2003, Nashville, TN.
28. General co-chair for the 5th International Symposium on Distributed Objects and Applications, November 3–7 2003, Catania, Sicily.
29. Program co-chair of the 3rd TAO Workshop, July 18, 2002, Arlington, VA.
30. Program co-chair for Middleware 2003, 4th IFIP/ACM/USENIX International Conference on Distributed Systems Platforms, June 16-20, 2003, Rio de Janeiro, Brazil.
31. Program co-chair for the 9th IEEE Real-time/Embedded Technology and Applications Symposium (RTAS), May 27-30, 2003, Washington, DC.
32. Area vice-chair and session chair for Middleware at the 23rd IEEE International Conference on Distributed Computing Systems (ICDCS), May 19-22nd, 2003, Providence, RI.
33. Program co-chair of the IEEE Workshop on LargeScale Real-Time and Embedded Systems, December 2, 2002, Austin, TX.
34. Program co-chair for the 4th International Symposium on Distributed Objects and Applications, October 28–November 1, 2002, Irvine, CA.
35. Co-organizer of the cross-agency Software Design and Productivity Coordinating Group Workshop on New Visions for Software Design and Productivity: Research and Applications, December 13-14, Nashville, TN.
36. Program co-chair for the 3rd International Symposium on Distributed Objects and Applications, September 18-20, 2001, Rome, Italy.
37. Co-organizer of the cross-agency Workshop on New Visions for Software Design and Productivity, April 18-19, 2000, Ballston, VA.
38. Area vice-chair and session chair for Middleware at the IEEE International Conference on Distributed Computing Systems, April 16-19, Phoenix, AZ, 2001.
39. Tutorial chair for the 6th USENIX Conference on Object-Oriented Technologies and Systems, January 27 - February 3, 2001, San Antonio, TX.
40. Co-chair of the OMG Workshop on Real-time and Embedded CORBA, in Reston, VA, July 24-27, 2000.
41. General chair of the IFIP/ACM International Conference Middleware 2000 in New York, April, 2000.

42. Tutorial chair for the 5th USENIX Conference on Object-Oriented Technologies and Systems, May 3-7, 1999, San Diego, CA.
43. Treasurer for the Fourth International Workshop on Object-oriented Real-time Dependable Systems (WORDS'99) January 27-29, 1999, Radisson Hotel, Santa Barbara, California, USA.
44. Tutorial chair for the 4th USENIX Conference on Object-Oriented Technologies and Systems, April 27-30, 1998, Santa Fe, New Mexico.
45. Co-chair of the mini-track on Engineering Client-Server Systems for the HICSS-31 conference, the Big Island of Hawaii - January 6-9, 1998.
46. Tutorial chair for the 3rd USENIX Conference on Object-Oriented Technologies and Systems, Portland, OR, June 1997.
47. Publicity chair for the 5th IEEE International Workshop on Object-Orientation in Operating Systems, IEEE TCOS and USENIX, Seattle, Washington, October 27-28, 1996.
48. Program chair for 3rd conference on Programming Languages of Programming, Allerton, IL, USA, September, 1996.
49. Program chair for the 2nd USENIX Conference on Object-Oriented Technologies, June 1996.

Professional Service and Advisory Positions

1. Member of the steering committee for the Software Product-Line Conference series.
2. Member of the Future Airborne Capabilities Environment (FACE) Advisory Board.
3. Vice-Chair of the Cyber Situation Awareness study for the Air Force Science Advisory Board.
4. Member of the Joint Tactical Radio System (JTRS) Tiger Team in support of the Assistant Secretary of the Army, Acquisition, Logistics, and Technology.
5. Member of the Air Force Science Advisory Board.
6. Member of the advisory board for the NSF-sponsored Repository for Model-Driven Development (ReMoDD) project at Colorado State University.
7. Member of the National Academics Committee on Advancing Software-Intensive Systems Producibility, chaired by Bill Scherlis from Carnegie Mellon University (CMU).
8. Member of the Engineering and Methods Technical Advisory Group (TAG) for the Software Engineering Institute at Carnegie Mellon University (CMU) from 2006 to 2009.
9. Member of the Ultra-Large-Scale (ULS) Systems study commissioned by the US Army and conducted at the Software Engineering Institute at Carnegie Mellon University (CMU).
10. Member of the Joshua group, which is an advisory board for the Air Force Research Lab (AFRL) in Rome, NY.

11. Member of the steering committee for the Distributed Objects and Applications conference series.
12. Member of the steering committee for the ACM/USENIX/IFIP Middleware conference series.
13. Member of the steering committee for EMSOFT 2002: Second Workshop on Embedded Software, Grenoble, France, October, 7–9th, 2002.
14. Member of the steering committee for EMSOFT 2001: First Workshop on Embedded Software, Lake Tahoe, California, October, 8th–10th, 2001.
15. Member of the Board of Directors for the Embedded Systems Consortium for Hybrid and Embedded Research (ESCHER).
16. Member of the NASA/JPL Mars Science Laboratory Mission Concept Review Board.
17. Chair of the subcommittee on Embedded and Hybrid Systems program for the National Science Foundation's 2003 Committee of Visitors in the Computer and Communications Research (C-CR) Division.
18. Co-chair of the Software Design and Productivity (SDP) Coordinating Group of the Federal government's multi-agency Information Technology Research and Development (IT R&D) Program, the collaborative IT research effort of the major Federal science and technology agencies. The SDP Coordinating Group formulates the multi-agency research agenda in fundamental software design.
19. One of the three founding members of the Scientific Advisory Board for the *International Symposium of Distributed Objects and Applications* conference series.
20. Member of the advisory board for Entera, which provides Internet content delivery systems based on ACE.
21. Invited to participate in the OO Working Group of the "Strategic Directions in Computing Research" workshop sponsored by ACM at MIT in June 1996.

Technical Program Committees

1. 10th ACM International Conference on Distributed and Event-based Systems, June 20 to June 24, 2016 in Irvine, CA.
2. First International Workshop on Science of Smart City Operations and Platforms Engineering (SCOPE), April 11, 2016, Vienna, Austria (Co-located with CPS Week).
3. 9th Dynamic Software Product Lines (DSLP) 2015 (held as part of SASO 2015) at MIT on September 21, 2015.
4. 13th International Conference on Advances in Mobile Computing and Multimedia (MoMM2015), Brussels, Belgium from 10-12 December 2015.
5. 13th IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC 2015, track on Cyber Physical Systems, Porto Portugal, October 21-23, 2015.
6. 35th IEEE International Conference on Distributed Computing Systems (ICDCS), June29 - July 2, 2015 in Columbus, Ohio, USA.

7. Fourth International Conference on Emerging Applications of Information Technology (EAIT) at Indian Statistical Institute, Kolkata, India, December 19-21, 2014.
8. The 20th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2014), Berlin, Germany, April 2014.
9. International Conference on Model-Driven Engineering and Software Development (MODELSWARD 2014), Lisbon, Portugal, 7-9 January, 2014.
10. 14th ACM/IFIP/USENIX International Middleware Conference (Middleware 2013), December 9-13, Beijing, China.
11. 32nd International Symposium on Reliable Distributed Systems (SRDS 2013), September 30-October 3, 2013 at Braga, Portugal.
12. 17th International Software Product Line Conference SPLC, Tokyo, Japan, 26-30 August 2013.
13. First International Workshop on Engineering Mobile-Enabled Systems, in conjunction with ICSE 2013, May 18-26th, 2013, San Francisco, CA.
14. International Conference on Model-Driven Engineering and Software Development (MODELSWARD 2013), Barcelona, Spain, 19-21 February, 2013.
15. ACM/USENIX/IFIP International Middleware conference, Montreal, Quebec, Canada, December 3-7, 2012.
16. 11th Workshop on Adaptive and Reflective Middleware, in conjunction with Middleware 2012 in Montreal, Quebec, Canada, December 3-7, 2012.
17. International Workshop on Real-Time and Distributed Computing in Emerging Applications (REACTION) 2012, San Juan, Puerto Rico, December 4, 2012, in co-location with the 33rd IEEE Real-Time Systems Symposium.
18. Third International Conference on Emerging Applications of Information Technology (EAIT) November 29 - December 01, 2012, Kolkata, India.
19. IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS), Las Vegas, USA, November 12 - 14, 2012.
20. 31st International Symposium on Reliable Distributed Systems (SRDS), 8th-11th October 2012. Irvine, California.
21. Sixth International Workshop on Dynamic Software Product Lines (DSPL), September 2 - 7, 2012, Salvador, Brazil.
22. 16th International Software Product Line Conference (SPLC 2012), Salvador, Brazil on 02-07 September 2012.
23. 5th International workshop UML and Formal Methods (UML&FM 2012), Paris, France, August 27-31, 2012.
24. UML&AADL 2012, July 18-20,2012, Ecole Normale Superieure, Paris, France.
25. 17th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS 2012), July 18-20,2012, Ecole Normale Superieure, Paris, France.
26. COMPSAC 2012 - Trustworthy Software Systems for the Digital Society, July 16-20, 2012, Izmir, Turkey.
27. Foundations Track of the 8th European Conference on Modelling Foundations and Applications (ECMFA 2012), Copenhagen, Denmark, 2-6th of July, 2012.

28. 24th International Conference on Software Engineering and Knowledge Engineering, Redwood City, California, USA, July 1-3, 2012.
29. 12th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS'12), Stockholm, Sweden, 13-16 June 2012.
30. 15th IEEE International Symposium on Object and component-oriented Real-time distributed Computing (ISORC), April 11-13, 2012, Shenzhen, China.
31. 23rd IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS 2011), Dallas, USA, December 14 to 16, 2011.
32. Fourth IEEE International Workshop on Real-Time Service-Oriented Architecture and Applications (RTSOAA 2011), December 12th–14th 2011, University of California, Irvine, CA.
33. ACM/IFIP/USENIX International Middleware Conference, Lisbon, Portugal, December 12th to 16th, 2011.
34. 9th International Conference on Advances in Mobile Computing and Multimedia (MoMM2011), Hue City, Vietnam, 05-07 December 2011.
35. Control Systems, Automation and Robotics track of the 3rd International Congress on Ultra Modern Telecommunications and Control Systems (ICUMT 2011), Hungary on October 5-7, 2011.
36. 15th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2011), August 29th - September 2nd, 2011, Helsinki, Finland.
37. 15th International Software Product Line Conference (SPLC 2011), Research/Experience Track, Munich, Germany, August, 22-26, 2011.
38. 15th International Software Product Line Conference (SPLC 2011), Industry Track, Munich, Germany, August, 22-26, 2011.
39. 2nd Workshop on Formal Methods in Software Product Line Engineering - Munich (Germany), August 2011.
40. 23rd International Conference on Software Engineering and Knowledge Engineering (SEKE2011), Miami Beach, USA, July 7-9, 2011.
41. 2nd International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems, July, 5th 2011, Porto, Portugal.
42. Fourth IEEE International workshop UML and Formal Methods, co-located with FM 2011, June 20th, 2011, Lero, Limerick, Ireland.
43. The Software Engineering and Data Engineering (SEDE 2011) conference, Las Vegas, Nevada, June 20-22, 2011.
44. 3rd International Workshop on Model-Driven Architecture and Modeling-Driven Software Development (MDA&MDSD 2011) in conjunction with the 6th International Conference on Evaluation of Novel Approaches to Software Engineering - ENASE 2011, Beijing Jiaotong University, 8-11, June 2011.
45. 11th International IFIP Conference on Distributed Applications and Interoperable Systems (DAIS 2011), Reykjavik, Iceland, June 6-9 2011.
46. Second Product LinE Approaches in Software Engineering (PLEASE) workshop, collocated with 33rd International Conference on Software Engineering, Waikiki, Honolulu, Hawaii, May 21-28, 2011.

47. 16th Annual IEEE International Conference on the Engineering of Complex Computer Systems (ICECCS), April 27th-29th, 2011 Las Vegas, NV, USA.
48. Sixth IEEE International workshop UML and AADL, in conjunction with ICECCS 2011, April 27th, 2011, Las Vegas, USA.
49. First International Workshop on Cyber-Physical Networking Systems (CPNS'2011), in conjunction with INFOCOM 2011, April 15, 2011, Shanghai, China.
50. 2nd Workshop on Model Based Engineering for Embedded System Design (M-BED 2011), colocated with the Design, Automation, and Test in Europe (DATE) conference, 14-18, March, 2011, Grenoble, France.
51. Second International Conference on Emerging Applications of Information Technology (EAIT 2011), February, 2011 at Kolkata, India.
52. Fifth International Workshop on "Variability Modeling of Software-intensive Systems" (VaMoS '11), January 27-29 2011 in Namur, Belgium.
53. 9th Workshop on Adaptive and Reflective Middleware (ARM 2010) November 27, 2010, Bangalore India, collocated with Middleware 2010.
54. The 22nd IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS 2010), November 8-10, 2010, Marina Del Ray, California.
55. International Conference on Software Engineering, Management, and Application (ICSEMA 2010) Kathmandu, Nepal, October 29th and 30th, 2010.
56. The MobiCPS 2010 workshop, held in conjunction with the 7th International Conference on Ubiquitous Intelligence and Computing (UIC2010), October 26-29, 2010 Xian, China.
57. Fourteenth IEEE International Enterprise Computing Conference (EDOC 2010), 25-29 October 2010, Vitoria, ES, Brazil.
58. Advances in Business ICT (ABICT) 2010 Workshop Wisla, Poland, October 18-20, 2010.
59. 3rd Workshop on Model Based Architecting and Construction of Embedded Systems (ACES-MB), held in conjunction with MoDELS 2010, Oslo, Norway, October 3-8, 2010.
60. 4th Dynamic Software Product Line Workshop held in conjunction with the 14th International Software Product Line Conference 2010, Jeju Island, South Korea, September 13-17, 2010.
61. TOOLS Europe 2010, Malaga, Spain, June 28 to July 2, 2010.
62. 22nd International Conference on Software Engineering and Knowledge Engineering (SEKE'2010), to be held July 1-3, 2010, Redwood City, California.
63. 13th International Symposium on Component Based Software Engineering (CBSE 2010), June 23-25 2010 in Prague, Czech Republic.
64. Sixth European Conference on Modelling Foundations and Applications (ECMFA), University of Pierre & Marie Curie, Paris, France. June 15-18, 2010.
65. 10th IFIP WG 6.1 International Conference on Distributed Applications and Interoperable Systems (DAIS), Amsterdam, The Netherlands, June 7-9, 2010.

66. The 11th OMG Real-time/Embedded CORBA workshop, Washington DC, May 24-26, 2010.
67. Industrial track at the 32nd International Conference on Software Engineering (ICSE 2010), Cape Town (South Africa), May 2-8, 2010.
68. Thirteeth International Conference on Business Information Systems (BIS 2010), Berlin, Germany, May 3-5 2010.
69. 1st International Workshop on Product LinE Approaches in Software Engineering, May 2nd, 2010, Cape Town, South Africa, held in conjunction with the 32nd International Conference on Software Engineering (ICSE 2010).
70. Workshop on Effective Multicasting for Future Critical Networked Systems (EMFINES 2010), at the Eighth European Dependable Computing Conference (EDCC), Valencia, Spain, April 28-30, 2010.
71. 1st Workshop on Model-Based Engineering for Embedded Systems Design, co-located with DATE 2010, March 12, 2010 in Dresden, Germany.
72. IEEE International Conference on Engineering of Complex Computer Systems (ICECCS 2010), Oxford 22-26, March 2010.
73. Special session on "Advanced Peer-to-Peer Protocols and Applications" at the Ninth IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN 2010) February 16-18, 2010 Innsbruck, Austria.
74. Fourth Variability Modelling of Software-intensive Systems (VaMoS '10) workshop, Linz, Austria - January 27-29, 2010.
75. 8th Workshop on Adaptive and Reflective Middleware (ARM'09), in collocation with the 10th ACM/IFIP/USENIX Middleware Conference, in Urbana Champaign, Illinois, November 30th, 2009.
76. Workshop committee for OOPSLA 2009, Orlando Floria, October 25-29, 2009.
77. The ARTIST 2nd International Workshop on Model Based Architecting and Construction of Embedded Systems (ACESMB 2009), in conjunction with the 12th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems (MODELS 2009), October 6th, 2009, Denver, Colorado.
78. The Thirteenth IEEE International EDOC Conference (EDOC 2009), 31 August - 4 September 2009, Auckland, New Zealand.
79. The 10th OMG Real-time/Embedded CORBA workshop, Washington DC, July 13–15, 2009.
80. The Software Engineering and Knowledge Engineering (SEKE'2009) conference, July 1-3, 2009, Boston, MA.
81. 12th International Symposium on Component-Based Software Engineering (CBSE 2009), East Stroudsburg University, Pennsylvania, USA, June 22-25, 2009.
82. The Second International Workshop on Cyber-Physical Systems (WCPS2009), held in conjunction with IEEE ICDCS 2009 in Montreal, Canada, June 22, 2009.
83. The Fifth European Conference on Model Driven Architecture Foundations and Applications (ECMDA), Gdansk, Poland, summer of 2009.

84. The 9th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS 2009) conference, Lisbon, Portugal, June 9-11, 2009.
85. The Fourth International Conference on COMmunication System softWAre and middlewaRE (COMSWARE), 15th - 19th June 2009, Trinity College Dublin, Ireland.
86. The UML&AADL Workshop, held in conjunction with ICECCS 2009 The fourteenth IEEE International Conference on Engineering of Complex Computer Systems June 02, 2009, Potsdam, Germany.
87. The 15th Real-time and Embedded Applications Symposium (RTAS) 2009, Track B, Real-time and Embedded Applications, Benchmarks and Tools, San Francisco, CA, United States, April 13 - 16, 2009.
88. Member of the ISORC 2009 advisory and publicity committee for ISORC 2009, March 17-20, 2009, Toyko, Japan.
89. the 13th International Software Product Line Conference (SPLC), August 24-28, 2009, San Francisco, CA.
90. the European Conference on Model Driven Architecture - Foundations and Applications 2009, University of Twente, Netherlands, June 2009.
91. The third workshop on "Variability Modelling of Software-intensive systems" (VaMoS'09), January 28-30 2009 in Sevilla, Spain.
92. the 1st Workshop on Software Reuse Efforts, November 27-29, 2008 Brazil.
93. the 7th Workshop on Adaptive and Reflective Middleware (ARM'08) in collocation with the 9th ACM/IFIP/USENIX Middleware Conference, Leuven, Belgium, December 1st 2008.
94. the Middleware 2008 9th International Middleware Conference, December 1-6, 2008, Leuven, Belgium.
95. the 11th Component-Based Software Engineering conference, Karlsruhe, Germany, October 14-17, 2008.
96. the ARTIST International Workshop on Model Based Architecting and Construction of Embedded Systems (ACESMB 2008), in conjunction with the 11th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems (MODELS 2008), Toulouse, September 29th, 2008.
97. the 6th Java Technology for Real-Time and Embedded Systems (JTRES) conference, Santa Clara, California, USA, 24-26 September, 2008.
98. the 12th IEEE International Enterprise Distributed Computing Conference (EDOC) (EDOC 2008), 15-19 September 2008, Munich, Germany.
99. the First Workshop on Analyses of Software Product Lines (ASPL'08), September 12, 2008 in Limerick, Ireland in conjunction with SPLC'08.
100. the 9th OMG Real-time/Embedded CORBA workshop, Washington DC, July 14–17, 2008
101. the 3rd International Conference on Software and Data Technologies, July 5-8, 2008, Porto, Portugal.
102. the 20th International Conference on Software Engineering and Knowledge Engineering (SEKE'08), Redwood City, California, USA, July 1-3, 2008.

103. the TOOLS EUROPE 2008 conference, June 30 to July 4, 2008 at ETH Zurich.
104. National Conference on Research & Development in Hardware & Systems (CSI-RDHS 2008), Computer Society of India Kolkata Chapter & CSI Division I (Hardware & Systems), June 20-21, 2008, Kolkata, India.
105. the First International Workshop on Cyber-Physical Systems, Beijing, China, June 17 - 20, 2008.
106. the ECMDA 2008 (Fourth European Conference on Model Driven Architecture Foundations and Applications) in Berlin, June 09 - 12, 2008.
107. the Distributed Applications and Interoperable Systems (DAIS), Oslo, Norway, June 4, 2008.
108. the 2nd International Workshop on Ultra-Large-Scale Software-Intensive Systems (ULSSIS 2008), May 10-11, 2008 Leipzig, Germany.
109. the Automotive Systems Track at the 30th International Conference on Software Engineering (ICSE), Leipzig, Germany, 10-18 May 2008.
110. the Real-Time and embedded Applications / Benchmarks track at the 14th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2008), St. Louis, MO, April 22-24, 2008.
111. the 3rd UML and AADL Workshop held in conjunction with the 13th IEEE International Conference on Engineering of Complex Computer Systems, Belfast, Northern Ireland, 31 March - 4 April 2008.
112. the ACM Programming for Separation of Concern track at SAC 2008, Fortaleza, Brazil, March 16 - 20, 2008.
113. the 6th edition of the International Workshop on Adaptive and Reflective Middleware, held in conjunction with Middleware 2007 in Newport Beach, California.
114. the IEEE/ACM/USENIX Middleware conference, November 2007.
115. the IASTED International Conference on Parallel and Distributed Computing and Systems, PDCS 2007, Cambridge, MA, USA from Nov 19-21, 2007.
116. the 9th International Symposium on Distributed Objects, Middleware, and Applications (DOA), Iberian peninsula and islands, Oct 28 - Nov 2, 2007.
117. Member of the Doctoral Symposium committee at OOPSLA 2007, Portland, OR October 21-25, 2007.
118. the International Symposium on Ambient Intelligence and Computing, October 2007, Korea.
119. the IEEE conference on Enterprise Distributed Object Computing (EDOC), Annapolis, MD, October 15-19, 2007.
120. the 5th Java Technology for Real-Time and Embedded Systems (JTRES), Vienna, Austria, 26-28 September, 2007.
121. the Workshop on Trade-Off analysis of Software Quality Attributes (TOSQA), collocated with the sixth joint meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering, Dubrovnik, Croatia, September 3-7, 2007.
122. the 2nd International Conference on Software and Data Technologies, July 22-25th 2007, Barcelona, Spain.

123. the Fourth IEEE International Conference on Web Services, Salt Lake City, UT, July 9-13, 2007.
124. the 10th International Component-Based Software Engineering (CBSE) Symposium, Boston, MA, July 9-11 2007.
125. the 8th OMG Real-time/Embedded CORBA workshop, Washington DC, July 9–12, 2007.
126. the International Conference TOOLS EUROPE 2007, Zurich, Switzerland on June, 24-28 2007.
127. the track on "Real-Time and Embedded Applications and Benchmarks" for the 13th IEEE Real-Time and Embedded Technology and Applications Symposium, Bellevue, WA, April 3 - April 6, 2007.
128. the Workshop on the Foundations of Interactive Computation (FInCo 2007), Braga, Portugal, March 24 - April 1, 2007.
129. the 15th International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS), Long Beach, California, 26-27 March, 2007.
130. the ACM Symposium on Applied Computing, Programming for Separation of Concerns track, Seoul, Korea, March 11 - 15, 2007.
131. the Workshop on Pervasive Computing Environments and Services (PCES 07), Naples, Italy, Feb 7-9, 2007.
132. the Minitrack on Components for Embedded and Real-time Systems at the 40th Hawaiian International Conference on System Sciences, January 3-6, 2007 at Waikoloa, Big Island, Hawaii.
133. the 13th Asia Pacific Software Engineering Conference (APSEC06), Bangalore, India, Dec 6-8, 2006.
134. the Real-time Middleware and Software Engineering track of the The 27th IEEE Real-Time Systems Symposium, December 5-8, 2006 Rio de Janeiro, Brazil.
135. the 2nd International Conference on Trends in Enterprise Application Architecture, November 29th to December 1st, 2006, Berlin, Germany.
136. the workshop on MOdel Driven Development for Middleware (MODDM), November 27, 2006, Melbourne, Australia.
137. the International Symposium on Distributed Objects and Applications (DOA), Montpellier, France, Oct 29 - Nov 3, 2006.
138. the "Library-Centric Software Design" (LCSD'06) workshop at the OOPSLA'06 conference in Portland, Oregon, October 22-26, 2006.
139. Judge for the Student Research Competition at OOPSLA 2006, Portland, OR, October 23-24, 2006.
140. the NSF Workshop On Cyber-Physical Systems, October 16 - 17, 2006, Austin, Texas.
141. the Models at Run-Time MaRT-06 workshop held at the MoDELS 2006 conference, Genova Italy, Oct. 2-6, 2006.
142. the MoDELS 2006 conference, Genova Italy, Oct. 2-6, 2006.
143. the 7th OMG Real-time/Embedded CORBA workshop, Washington DC, July 11–14, 2006.

144. the European Conference on Object-Oriented Programming, Nantes, France, July 3-7, 2006.
145. the 9th International Symposium on Component-Based Software Engineering (CBSE 2006), Malardalen University, Sweden, June 29th-1st July 2006.
146. the 28th International Conference on Software Engineering (ICSE 28), May 24-26, 2006, Shanghai, China.
147. the 14th International Workshop on Parallel and Distributed Real-Time Systems, April 25-26, 2006, Island of Rhodes, Greece.
148. the 9th IEEE International Symposium on Object-oriented Real-time Distributed Computing, April 24-26, 2006, Gyeongju, Korea.
149. the Automotive Software Workshop San Diego (ASWSD 2006), University of California, San Diego, March 15-17, 2006.
150. the C++ Connections: 20 Years of C++ conference, Nov 7-11, 2005, Mandalay Bay, Las Vegas, NV.
151. the Conference on Distributed Objects and Applications (DOA 2005), Oct 31 - Nov 4, 2005, Agia Napa, Cyprus.
152. the 20th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOSPLA), October 16-20, 2005, San Diego, CA, USA.
153. the 6th International Conference on Middleware (Middleware'2005), October, 2005, Grenoble, France.
154. the 2005 Monterey Workshop on Networked Systems, Laguna Beach, California, September 22-24, 2005.
155. The 12th Pattern Language of Programs (PLoP 2005), September 7-10, 2005, Allerton Park, Monticello, Illinois, USA.
156. the 14th IEEE International Symposium on High-Performance Distributed Computing (HPDC-14), Research Triangle Park, North Carolina, July 27, 2005.
157. the 5th International Workshop on Software and Performance (WOSP 2005), Palma de Mallorca, Spain, July 11-15, 2005.
158. the 6th OMG Real-time/Embedded CORBA workshop, Washington DC, July 11–14, 2005.
159. the 5th IFIP WG 6.1 International Conference on Distributed Applications and Interoperable Systems (DAIS 2005), June 15-17, 2005, Athens, Greece.
160. the International Conference on Autonomic Computing (ICAC 2005), Seattle, WA, June 2005.
161. the International Symposium on Component-Based Software Engineering (CBSE), co-located with the International Conference on Software Engineering (ICSE), May 14-15, 2005, St. Louis, MO.
162. the Foundations of Interactive Computation (FINCO'05) Workshop, Saturday, 9 April 2005, in Edinburgh, Scotland.
163. the Embedded Applications track of the IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS) 2005, San Francisco, California, March 2005.

164. the "Programming for Separation of Concerns" track at Symposium on Applied Computing (SAC 2005), Santa Fe, New Mexico, March 2005.
165. the 12th International Symposium on the Foundations of Software Engineering, November 6th, 2004, Newport Beach, California.
166. the Conference on Distributed Objects and Applications (DOA 2004), October 25-29, 2004 in Cyprus, Greece.
167. the 2nd International Workshop on Java Technologies for Real-Time and Embedded Systems (JTRES), October 25-29, 2004, Larnaca, Cyprus.
168. the 3rd Workshop on Reflective and Adaptive Middleware (RM2004), October 19, 2004, Toronto, Ontario, Canada.
169. the Middleware 2004 5th IFIP/ACM/USENIX International Conference on Distributed Systems Platforms, October 18-22, 2004, Toronto, Canada.
170. the 4th TAO+CIAO Workshop, Arlington, VA, July 16, 2004.
171. the DARPA Workshop on Java in Real-Time and Embedded Defense Applications, Arlington, VA, July 13, 2004.
172. the OMG Real-time/Embedded CORBA workshop, Crystal City, VA, July 12–15, 2004.
173. the ECOOP 2004 conference, June 14-18, 2004, Oslo, Norway.
174. the Middleware track of the 24th IEEE International Conference on Distributed Computing Systems (ICDCS), May 23-26, 2004, Tokyo, Japan.
175. the 2nd International Workshop on Remote Analysis and Measurement of Software Systems (RAMSS), Edinburgh, Scotland, UK, May 24, 2004.
176. Aspect-Oriented Software Development conference, Lancaster, England, March 22-26, 2004.
177. the SPIE/ACM Conference on Multimedia Computing and Networking, January 21-22, 2004 Santa Clara, California.
178. the Real-time Systems Symposium (RTSS), Cancun, Mexico, December 3-5, 2003.
179. the 4th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS), Paris - France November 17-21, 2003.
180. the International Workshop on Java Technologies for Real-Time and Embedded Systems (JTRES), November 3-7, 2003, Catania, Sicily, Italy.
181. the Domain Driven Development track at the OOPSLA 2003 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, October 26-30, 2003, Anaheim, California, USA.
182. the OOPSLA 2003 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, October 26-30, 2003, Anaheim, California, USA.
183. External reviewer for the 2nd Generative Programming and Component Engineering (GPCE '03) conference, Erfurt, Germany, September 22-25, 2003.
184. the OMG Real-time/Embedded CORBA workshop, Crystal City, VA, July 14–17, 2003.
185. the The 2nd Workshop on Reflective and Adaptive Middleware, Rio de Janeiro, Brazil, June 17, 2003.

186. the ACM SIGPLAN 2003 Conference on Programming Language Design and Implementation (PLDI), San Diego, California, June 9 - 11, 2003.
187. the 1st International Workshop on Remote Analysis and Measurement of Software Systems (RAMSS), Portland, Oregon, May 9, 2003.
188. External reviewer for the 17th International Parallel and Distributed Processing Symposium, April 22–26, 2003, Nice, France.
189. the ACM International Conference on Aspect-Oriented Software Development, March 17 - 21, 2003, Boston, MA.
190. the SPIE/ACM Conference on Multimedia Computing and Networking, Santa Clara, California, January 29–31, 2003.
191. the International Workshop on Product Line Engineering The Early Steps: Planning, Modeling, and Managing (PLEES '02), Seattle, WA, November 5, 2002.
192. the 8th IEEE Real-Time and Embedded Technology and Application Symposium (RTAS), San Jose, CA, September 24-27, 2002.
193. the 9th Conference on Pattern Language of Programs, Allerton Park, IL, September 8-12, 2002.
194. the Workshop on Dependable Middleware-Based Systems, held as a part of DSN 2002, Washington, D.C., June 23-36, 2002.
195. the 2nd TAO Workshop, Arlington, VA, July 19, 2002.
196. the OMG Real-time/Embedded CORBA workshop, Crystal City, VA, July 15–18, 2002.
197. the 16th European Conference on Object-Oriented Programming, University of Malaga, Spain June 10-14, 2002.
198. the Tenth International Workshop on Quality of Service (IWQoS), May 15-17, 2002, Miami Beach, Florida.
199. the International Symposium on Object-Oriented Real-time Distributed Computing (ISORC), Washington DC, April 29 – May 1, 2002.
200. the Seventh IEEE International Workshop on Object-oriented Real-time Dependable Systems (WORDS 2002), January 7-9, 2002, San Diego, CA.
201. the International Workshop on Multimedia Middleware October 5th, 2001, Ottawa, Canada.
202. the OMG Workshop on Real-time and Embedded CORBA, in Reston, VA, June 4-6, 2001.
203. the USENIX 2001 conference, Boston, MA, June 25-30, 2001.
204. the International Symposium on Object-oriented Real-time Distributed Computing (ISORC), May 2-4, Magdenburg, Germany, 2001.
205. the 6th USENIX Conference on Object-Oriented Technologies and Systems, January 27 - February 3, 2001, San Antonio, TX.
206. External reviewer for OOPSLA 2000, Minneapolis, MN, October 2000.
207. the 3rd IFIP International Conference on Trends towards a Universal Service Market (USM'2000), September 12-14, 2000.
208. the International Symposium on Distributed Objects and Applications (DOA '00), OMG, Antwerp, Belgium, September 2000.

209. the ACM SIGCOMM 2000, Stockholm, Sweden, August 30 to September 1st, 2000.
210. the Pattern Languages of Programming (PLoP) conference, Monticello, Illinois, August, 2000.
211. the 9th IEEE International Conference on High-Performance Distributed Computing, August, 2000.
212. the "International Workshop on Software Engineering for Parallel and Distributed Systems" (PDSE 2000), at the 22nd International Conference on Software Engineering (ICSE-2000), in Limerick, Ireland in June, 2000.
213. the 6th IEEE Real-Time Technology and Application Symposium (RTAS), May 17-19, 2000, Washington DC, USA.
214. the 1999 ACM OOPSLA conference, Denver, Colorado, November 1-5, 1999.
215. the IFIP Sixth International Workshop on Protocols For High-Speed Networks (PfHSN '99), Wednesday August 25 – Friday August 27, 1999 Salem, MA.
216. the 1999 IEEE Real-Time Technology and Applications Symposium (RTAS99), Vancouver, British Columbia, Canada, June 2-4, 1999.
217. the 5th USENIX Conference on Object-Oriented Technologies and Systems, May 3-7, 1999, San Diego, CA.
218. Technical workshop committee for the International Software Architecture workshop, ACM SIGSOFT's FSE9 conference in Orlando FL, November 1-5, 1998.
219. the workshop on Software and Performance (WOSP98), Santa Fe, New Mexico, Oct 12-16 1998.
220. the IFIP International Conference on Distributed Systems Platforms and Open Distributed Processing: Middleware '98. September 15-18 1998, The Lake District, England.
221. the TOOLS USA'98 conference. Santa Barbara, California, August 3 - 7, 1998.
222. the IEEE High Performance Distributed Computing conference, Chicago, IL, July 28-31, 1998.
223. 12th European Conference on Object-Oriented Programming, Brussels, Belgium, July 20 - 24, 1998.
224. the 3rd EuroPLoP conference, Kloster Irsee, Germany, July 9-11, 1998.
225. the IEEE International Conference on Configurable Distributed Systems (ICCDS '98), Annapolis, MD, May 4-6, 1998.
226. the IEEE IWQoS '98 in Napa Valley, CA, May 18-20, 1998.
227. the 4th USENIX Conference on Object-Oriented Technologies and Systems, April 26-29, 1998, Santa Fe, New Mexico.
228. the 3rd International Workshop on Software Engineering for Parallel and Distributed Systems, at the 20th International Conference on Software Engineering (ICSE-20), in April 20-21, Kyoto, Japan.
229. the IEEE Conference on Open Architectures and Network Programming, April 3-4, 1998, San Francisco, CA.

230. the Workshop on Middleware for Real-Time Systems and Services, held in conjunction with IEEE Real-time Systems Symposium, December 2nd, San Francisco, California.
231. the Open Signaling for ATM, Internet and Mobile Networks. October 6th and 7th, 1997, Columbia University, New York, NY.
232. the 24*th* International Conference on Technology of Object-Oriented Languages and Systems (TOOLS Asia '97). Beijing, China, September 22 - 25, 1997.
233. the 4*th* Pattern Languages of Programming conference, Allerton Park, Illinois, September 3-5, 1997.
234. the 3*rd* USENIX Conference on Object-Oriented Technologies and Systems, Portland, June 16-19th 1997.
235. Session chair of the Patterns technical paper session at ECOOP '97, June 13th, 1997.
236. the 1997 European Conference on Object-Oriented Programming (ECOOP), June 9-13, 1997, Jyvskyl, Finland.
237. Chair of the technical session on "Distributed Object Computing" for the IFIP/IEEE Fifth International Workshop on Quality of Service (IWQoS '97).
238. the 2*nd* International Workshop on Software Engineering for Parallel and Distributed Systems, at the 19*th* International Conference on Software Engineering (ICSE-19) Sheraton Boston Hotel and Towers, Boston, Massachusetts, USA, May 19 and 20, 1997.
239. the 3*rd* USENIX Conference on Object-Oriented Technologies and Systems, Portland, 1997.
240. the 5*th* IEEE International Workshop on Object-Orientation in Operating Systems, IEEE TCOS and USENIX, Seattle, Washington, October 27-28, 1996.
241. the 1997 ACM SIGCOMM conference, Cannes, French Riviera, France, September 1997.
242. the 1997 IEEE INFOCOM conference, Kobe, Japan, April 1997.
243. the 1996 IEEE INFOCOM conference, San Francisco, CA, USA, March 24-28, 1996.
244. the 1995 IEEE INFOCOM conference, Boston, Massachusetts, USA, April, 1995.
245. the 3*rd* IEEE workshop on Architecture and Implementation of High Speed Communication Subsystems (HPCS '95), held in Mystic, Connecticut, August 1995.
246. the 8*th* IFIP International Working Conference on Upper Layer Protocols, Architectures, and Applications, held in Barcelona, Spain, June 1 to 3, 1994.

Workshops and Panels Organized

1. Co-organized the 1st International Workshop on Data Dissemination for Large scale Complex Critical Infrastructures (DD4LCCI 2010), at the Eighth European Dependable Computing Conference, Valencia, Spain, April 28-30, 2010.
2. Co-organized the OOPSLA Jeopardy panel at OOPSLA 2009, Orlando Floria, October 25-29, 2009.
3. Co-organized a workshop entitled First International Workshop on Software Technologies for Ultra-Large-Scale (ULS) Systems at 29th Int. Conference on Software Engineering, May 20-29th, Minneapolis, MN, 2007.
4. Co-organized a session on architectures, platforms, and standards for QoS-enabled dissemination at the Systems and Information Interoperability Meeting, Oct 25-27, 2006 at the Minnowbrook Conference Center, Blue Mountain Lake, NY.
5. Co-organized a workshop entitled "Breathturn: Ultra Large Scale Systems" at OOPSLA 2006, October 26, 2006, Portland, OR.
6. Co-chair of the NSF workshop on open-source Middleware for Distributed Real-time and Embedded Systems, 7th OMG Real-time/Embedded CORBA workshop, Arlington, VA, July 10–13, 2006.
7. Organized and led a session on architectures, platforms, and standards for real-time tactical information management at the Systems and Information Interoperability Meeting, Oct 18-21, 2005 at the Minnowbrook Conference Center, Blue Mountain Lake, NY.
8. Co-organizer of the technical workshops program at OOPSLA 2005, San Diego, October 16th-20, 2005.
9. Co-organizer for the MODELS 2005 workshop on "MDD for Software Product-lines: Fact or Fiction? ," October 2, 2005, Jamaica.
10. Co-organizer of the OOPSLA '02 workshop on "Patterns in Distributed Real-Time and Embedded Systems", Seattle, WA, November, 2002.
11. Co-organizer of the OOPSLA '01 workshop on "Towards Patterns and Pattern Languages for OO Distributed Real-time and Embedded Systems" Tampa Bay, FL, October 14, 2001.
12. Organizer and chair of a panel on real-time extensions to OO middleware, OPENSIG Fall '97 workshop on Open Signaling for ATM, Internet and Mobile Networks Columbia University, October 6-7 1997, New York, NY.
13. Co-organizer of a workshop for the 1997 European Conference on Object-Oriented Programming entitled CORBA: Implementation, Use, and Evaluation, Jyvaskyla, Finland, June 10th, 1997.
14. Organizer and chair of a panel on "QoS and Distributed Systems Platforms" for the IFIP Fifth International Workshop on Quality of Service (IWQoS '97), May 22-24th, 1997, Columbia University, New York.
15. Co-organizer of the OOPSLA '95 workshop on "Patterns for Concurrent, Parallel, and Distributed OO Systems."
16. Co-facilitator of the ECOOP '95 workshop workshop on Pattern Languages of Object-Oriented Programs, Aarhus, Denmark, August 1995.

Reviewer for Professional Submittals

Reviewed papers for the following journals, conferences, books, and grant review processes:

1. *IEEE Software*, Special Issue on Next Generation Mobile Computing, edited by James Edmondson et al., 2013.
2. *Software Testing in the Cloud*, edited by Scott Tilley, 2012.
3. Elsevier Information & Software Technology special issue on Software Reuse and Product Lines, 2012.
4. The 2010 Military Communications Conference, Cyber Security and Network Management, San Jose, CA, October 31-November 3, 2010.
5. *Model-Driven Domain Analysis and Software Development: Architectures and Functions*, edited by Janis Osis and Erika Asnina, 2010.
6. Reviewer for the book "Patterns for Parallel Software Design," by Jorge L. Ortega Arjona, Wiley, 2010.
7. Special Issue on Industrial Applications of Aspect Technology for the journal Transactions on Aspect-Oriented Software Development (TAOSD), 2009.
8. *Software Engineering for Self-Adaptive Systems*, edited by Betty H. C. Cheng, Rogerio de Lemos, Holger Giese, Paola Inverardi, and Jeff Magee, Springer, 2009.
9. Special issue on Service Oriented Computing for the ACM Transactions on the Web journal, 2008.
10. Special Issue in Software Reuse: Methods, Processes, Tools and Experiences for the Journal of the Brazilian Computer Society (JBCS), 2007
11. Designing Software-Intensive Systems: Methods and Principles book, 2008
12. Special issue on Patterns for the IEEE Software, 2007
13. IEEE Internet Computing Magazine, 2006.
14. IEEE Transactions on Parallel and Distributed Systems, 2004
15. International Journal of Software Process: Improvement and Practice Special issue - Software Variability: Process and Management
16. IEEE Internet Computing Magazine
17. 2004 NSF NSG panel
18. IEEE Transactions on Parallel and Distributed Computing special issue on Middleware, 2003
19. 2003 NSF ITR panel
20. 2002 NSF CAREER panel
21. IEEE Internet Computing Magazine, 2002
22. NIST Competence Proposals, May 2002
23. DARPA MoBIES program, May 2002
24. DARPA NEST program, May 2002
25. DARPA DASADA program, April 2002
26. Elsevier Journal of Systems and Software Special Issue on Software Architecture: Engineering Quality Attributes, 2002
27. IEEE Communications Magazine, Evolving Communications Software: Techniques and Technologies, 2001
28. DARPA Network Embedded Software Technology (NEST) program, 2001

29. DARPA Software Enabled Control (SEC) program, 2000
30. IEEE Concurrency magazine, Object-Oriented Systems Track, 1999
31. IEEE Journal on Selected Areas in Communications special issue on "Service Enabling Platforms for Networked Multimedia Systems," 1999
32. IEEE Journal of Communications and Networks, 1999
33. Reviewer for the 4th Pattern Languages of Programming Design book published by Addison Wesley
34. The International Journal of Time-Critical Computing Systems, special issue on Real-time Middleware, edited by Wei Zhao
35. Next Generation Internet (NGR) networking research review panel, October 1998
36. IEE Transactions on Software Engineering, special issue on Configurable Distributed Systems
37. Theme issue on Symbolic Modeling in Practice for the Communications of the ACM
38. "Multimedia DBMS and the WWW" Minitrack at the 32nd Hawaii International Conference on System Sciences, 1999
39. "Dependable Distributed Systems" Minitrack at the 32nd Hawaii International Conference on System Sciences, 1999
40. IEEE Computer special issue on "Design Challenges for High-Performance Network Interfaces," 1998
41. 1998 NSF Experimental Software Systems review panel.
42. ACM SIGMetrics Conference, 1998
43. ACM Transactions on Software Engineering Methods
44. Special Issue on Patterns and Pattern Languages for the journal of Theory and Practice of Object Systems, (Stephen P. Berczuk, Editor), John Wiley and Sons, 1995
45. Special Issue of Computer Communications on Building Quality of Service into Distributed Systems
46. IEEE Communications Magazine
47. IEEE/ACM Journal of Transactions on Networking
48. Communications of the ACM
49. IEE/BCS Distributed Systems Engineering Journal
50. Software Practice and Experience, John Wiley and Sons
51. 1998, 1997, and 1996 NSF networking program
52. 1996 NSF software engineering and programming languages CAREER panel
53. 1994 California MICRO (Microelectronics Innovation Computer Research Opportunity) engineering computer network grant review process
54. IEEE Conference on Parallel and Distributed Computing Systems, 1994
55. IEEE International Conference on Computer Communications and Networks, 1994
56. IEEE INFOCOM conference, 1994
57. 1993 NASA Applied Information Systems Research grant review process
58. 1992 California MICRO (Microelectronics Innovation Computer Research Opportunity) engineering computer network grant review process

59. *7th IFIP International Conference on Upper Layer Protocols, Architectures, and Applications*, 1992
60. The 1992 Special Issue on Measurement for IEEE Journal Transactions on Software Engineering

Patents

1. US patent 7,523,471 – "Interpretive network daemon implemented by generic main object," in conjunction with Karlheinz Dorn, Dieter Quehl, Detlef Becker, and Christian Scharf of SIEMENS Medical Engineering, Erlangen, Germany, 2009.

Theses Supervised

- *Doctoral and Masters Committees Chaired*
    1. Co-chair of the doctoral dissertation defense for James Edmondson, March 2012.
    2. Co-chair of the doctoral topic defense for James Edmondson, December 2011.
    3. Co-chair of the doctoral dissertation defense for Will Otte, November 2011.
    4. Chair of the doctoral dissertation defense for Brian Dougherty, March 2011.
    5. Chair of the doctoral topic defense for Brian Dougherty, June 2010.
    6. Chair of the masters defense for Pooja Varshneya, May 2010.
    7. Chair of the doctoral topic defense for Nilabja Roy, March 2010.
    8. Chair of doctoral topic defense for Joe Hoffert, November 2009.
    9. Chair of the doctoral dissertation defense for Jai Balasubramanian, September 2009.
    10. Chair of masters defense for Friedhelm Wolf, March 2009.
    11. Chair of the doctoral dissertation defense for Nishanth Shankaran, October 2008.
    12. Chair of the doctoral dissertation defense for Jules White, October 2008.
    13. Chair of doctoral dissertation defense for Gan Deng, December 2007.
    14. Chair of doctoral dissertation defense for Krishnakumar Balasubramanian, September 2007.
    15. Chair of the doctoral topic defense for Nishanth Shankaran, April 2007.
    16. Chair of doctoral topic defense for Krishnakumar Balasubramanian, March 2006.
    17. Chair of doctoral topic defense for Gan Deng, March 2006.
    18. Chair of final doctoral dissertation defense for Arvind Krishna, December 2005.
    19. Chair of MS thesis committee for Emre Turkay, summer 2005.
    20. Chair of doctoral topic defense for Arvind Krishna, summer 2005.
    21. Chair of MS thesis committee for Ossama Othman, December, 2002.
    22. Chair of doctoral dissertation committee for Carlos O'Ryan, May, 2002.

23. Chair of dissertation topic defense committee for Carlos O'Ryan, September, 2001.
24. Chair of masters committee for Nagarajan Surendran, August, 1999.
25. Chair of masters committee for Alexander Babu Arulanthu, July, 1999.
26. Chair of oral exam committee for Chris Gill, June, 1999.
27. Chair of doctoral exam committee for Andy Gokhale, May, 1998.
28. Chair of masters exam committee for Sumedh Mungee, May, 1998.
29. Chair of masters exam committee for Sergio Flores, May, 1998.
30. Chair of masters committee for Prashant Jain, June 1997.
31. Chair of doctoral topic defense for James Hu, February 1997.
32. Chair of masters committee for Tim Harrison, February 1997.
33. Chair of doctoral topic defense committee for Andy Gokhale, October, 1996.

- *Doctoral and Masters Committees Member*
  1. Served on the doctoral dissertation defense for Faruk Caglar, July 2015
  2. Served on the doctoral dissertation defense for Wei Yan, May 2015.
  3. Served on the doctoral dissertation defense for Kyoungho An, March 2015.
  4. Served on the MS thesis committee for Songtao Hei, March 2015.
  5. Served on the MS thesis committee for Meng Wang, March 2015.
  6. Served on the doctoral dissertation defense for Sean Hayes, January 2015.
  7. Served on the doctoral dissertation defense for Hamilton Turner, November 2014.
  8. Served on the doctoral topic defense for Faruk Caglar, November 2014.
  9. Served on the doctoral topic defense for Hamilton Turner, February 2014.
  10. Served on the doctoral dissertation defense for Fan Qui, February 2014.
  11. Served on the doctoral dissertation defense for Xiaowei Li, May 2013.
  12. Served on the doctoral topic defense for Fan Qiu, April 2013.
  13. Served on the doctoral dissertation defense for Janos Mathe, August 2012.
  14. Served on the doctoral dissertation defense for Tripti Saxena, July 2012.
  15. Served on the doctoral dissertation defense for Akshay Dabholkar, April 2012.
  16. Served on the doctoral topic defense for Xiawei Li, March 2012.
  17. Served on the doctoral topic defense for Janos Mathe, August 2011.
  18. Served on the doctoral dissertation defense for Liang Dai, April 2011.
  19. Served on the doctoral dissertation defense for Daniel Balasubramanian, March 2011.
  20. Served on the doctoral topic defense for Will Otte, February 2011.
  21. Served on the doctoral topic defense for Akshay Dabholkar, February 2011.
  22. Served on the doctoral dissertation defense for Joe Hoffert, February 2011.
  23. Served on the doctoral topic defense for Tripti Saxena, January 2011.

24. Served on the doctoral dissertatin defense for Nilabja Roy, November 2010
25. Served on the doctoral topic defense for Daniel Balasubramanian, October 2010.
26. Served on the doctoral dissertation defense for Sumant Tambe, September 2010.
27. Served on the doctoral topic defense for Sumant Tambe, April 2010.
28. Served on the doctoral dissertation defense for John Kinnebrew, March 2010.
29. Served on the doctoral dissertation defense for Shanshan Jiang, November 2009.
30. Served on the doctoral dissertation defense for James Hill, March 2009.
31. Served on the doctoral topic defense for James Hill, October 2008.
32. Served on the doctoral topic defense for Jai Balasubramanian, August 2008.
33. Served on the doctoral topic defense for Liang Dai, December 2008.
34. Served on the doctoral topic defense for Shanshan Jiang, November 2008.
35. Served on the doctoral topic defense for Jules White, April 2008.
36. Served on the doctoral topic defense for Amogh Kavimandan, February 2008.
37. Served on the doctoral dissertation defense for Amogh Kavimandan, November 2008.
38. Served on the doctoral topic defense for Amogh Kavimandan, February 2008.
39. Served on the doctoral dissertation defense for Michael Stal, University of Groningen, March 2007.
40. Served on the doctoral topic defense for Karlkim Suwanmongkol, fall 2004.
41. Served on the doctoral dissertation topic defense committee for Aditya Agrawal, July, 2004.
42. Served on the doctoral dissertation defense for Angelo Corsaro, July 2004.
43. Served on the doctoral dissertation defense for Nanbor Wang, April 2004.
44. Served on the doctoral topic defense for Angelo Corsaro, October 2003.
45. Served on the doctoral dissertation defense committee for Jonathan Sprinkle, July, 2003.
46. Served on the doctoral dissertation topic defense committee for Aditya Agrawal, June, 2003.
47. Served on masters committee for Kirk Kelsey, March 2003.
48. Served on the dissertation topic defense committee for Jonathan Sprinkle, February, 2003.
49. Served as external examiner for Bob Jolliffe's masters thesis Department of Computer Science, University of South Africa, March, 2003.
50. Served on the doctoral dissertation committee for Irfan Pyarali, December, 2001.

51. Served on the doctoral dissertation committee for Chris Gill, December, 2001.
52. Served as external examiner for Daniel Heggander's Ph.D. dissertation in the Department of Software Engineering and Computer Science at Blekinge Institute of Technology, Sweden, September, 2001.
53. Served as external examiner for Mohammad Radaideh's masters thesis in the Electrical Engineering department at McMaster's University, Canada, Winter 2000.
54. Served as external examiner for David Holmes' Ph.D. dissertation in the information and computer sciences department at Macquarie University, Sydney, Fall 1999.
55. Served on final doctoral dissertation committee for Priya Narasimhan, August, 1999.
56. Served on the doctoral final dissertation defense for Christo Papadopoulos, August, 1999.
57. Served on dissertation topic defense for Michael Plezbert, February, 1999.
58. Served on masters committee for Craig Nauman, February, 1999.
59. Served on the doctoral exam committee for Chuck Cranor, July, 1998.
60. Served on masters exam committee for Mihai Tutunaru, April, 1998.
61. Served on the doctoral exam committee for Michael Plezbert, June, 1997.
62. Served on masters committee for Todd Rogers, June 1997.
63. Served on masters committee for Robert Engel, January 1997.
64. Served on committee for final doctoral dissertation defense of R. Gopalakrishnan, November, 1996.
65. Served on committee for final doctoral dissertation defense of Lorrie Cranor, September, 1996.
66. Served on the doctoral dissertation topic proposal committee for Christos Papadopoulos July, 1995.
67. Served on the doctoral dissertation topic proposal committee for Charles Cranor December, 1994.
68. Served on oral exam committee for Andy Gokhale December, 1994.
69. Served on the doctoral dissertation proposal committee for Lorrie Cranor, December, 1994.
70. Served on the doctoral final dissertation defense committee for Donald Wilcox, November, 1994.
71. Served on masters committee for Madhavapeddi Shreedhar, September, 1994
72. Served on the doctoral dissertation topic proposal committee for R. Gopalakrishnan, September, 1994.

- *Doctoral Student Advisees and Co-Advisees*
  1. Mike Walker (USA)
- *Graduated PhD Students*
  1. Jaiganesh Balasubramanian, Ph.D., 2009, Citigroup, NY, NY.
  2. Krishnakumar Balasubramanian, Ph.D., 2007, Mathworks, Boston, MA.

3. Angelo Corsaro, Ph.D. 2004, PrismTechnologies, Parise France.
4. Gan Deng, Ph.D., 2007, Citigroup, Charleston, SC.
5. Brian Dougherty, Ph.D. 2011, Optio Labs, Nashville, TN.
6. James Edmondson, Ph.D., 2012, Member of the Technical Staff, Software Engineering Institute, Pittsburgh, PA.
7. Chris Gill, Ph.D. 2001, Professor, Washington University, St. Louis, MO.
8. Andy Gokhale, Ph.D. 1998, Associate Professor, Vanderbilt University, Nashville, TN.
9. James Hill, Ph.D., 2009, Associate Professor, Indiana University, Purdue University, Indianapolis.
10. Joe Hoffert, Ph.D. 2011, Assistant Professor, Indiana Wesleyan University, IN.
11. John Kinnebrew, Ph.D., 2010, Institute for Software Integated Systems, Nashville, TN.
12. Arvind Krishna, Ph.D. 2005, Qualcomm, San Diego, CA.
13. Irfan Pyarali, Ph.D. 2001, CitiGroup, New Jersey.
14. Nilabja Roy, Ph.D. 2011, Research Scientist, Institute for Software Integrated Systems, Nashville, TN.
15. Carlos O'Ryan, Ph.D., 2002, Google, NY, NY.
16. Nishanth Shankaran, Ph.D., 2008, LinkedIn, Silicon Valley, CA.
17. Nanbor Wang, Ph.D. 2004, Research Scientist, VM Ware, San Maeto, CA.
18. Jules White, Ph.D. 2008, Assistant Professor, Vanderbilt University.

- *Graduated Masters and Ugrad Students*
  1. Alexander Babu Arulanthu, MS 1999, Sylantro, Campbell, CA.
  2. Everett Anderson, BS 1998, Sun, Mountain View, CA.
  3. Shawn Atkins, BS 1998, Lucent, Columbus, OH.
  4. Matt Braun, BS 1998.
  5. Darrell Brunsch, BS 1999, Microsoft, Redmond, WA.
  6. George Edwards, BS 2004, Ph.D. student at University of Southern California.
  7. Sergio Flores-Gaitan, MS 1998, Microsoft, Redmond, WA.
  8. Priyanka Gontla, MS 2000, UBS, Irvine, CA.
  9. Pradeep Gore, MS 2000, OOMWorks, New Jersey.
  10. Tim Harrison, MS 1997, Mayasoft, Palo Alto, CA.
  11. Prashant Jain, MS 1997, IBM Research, India.
  12. Vishal Kachroo, MS 1999, Stentorsoft, CA.
  13. Michael Kircher, BS 1998, Siemens CT, Munich, Germany.
  14. Yamuna Krishnamurthy, MS 2000, OOMWorks, New Jersey.
  15. Tao Lu, MS 2003, Trading Technologies, Chicago, IL.
  16. Sumedh Mungee, MS 1998, Fujitsu, Santa Clara, CA.
  17. Bala Natarajan, MS 2000, Veritas, India.
  18. Kirthika Parameswaran, MS 2000, Telcordia, Piscataway, NJ.
  19. Stoyan Paunov, MS 2006, working at Bloomberg, NYC.
  20. Ossama Othman, MS 2002, independent consultant, Portland, OR.
  21. Marina Spivak, MS 2000, AT Desk, Charleston, SC.
  22. Nagarajan Surendran, MS 1999, Sylantro, Campbell, CA.

23. Emre Turkay, MS 2005, Turkey.
24. Pooja Varshneya, May 2010, Zircon Computing, Wayne, NJ.
25. Seth Widoff, BS 1998, independent consultant, San Francisco, CA.
26. Ming Xiong, MS 2007, AT Desk, Charleston, SC.
27. Violetta Vylegzhanina, Optio Labs, Nashville TN.

- *Former Staff*
  1. Chris Cleeland, OCI, St. Louis, MO.
  2. Ray Klefstad, Research Assistant Professor, University of California, Irvine.
  3. Boris Kolpackov, Independent Consultant, South Africa.
  4. Fred Kuhns, Research Associate, Washington University, St. Louis, MO.
  5. David Levine, Director of Engineering, CombineNet, Inc, Pittsburgh, PA.
  6. Will Otte, Institute for Software Integrated System
  7. Jeff Parsons, Optio Labs
  8. Jules White, Assistant Professor, Vanderbilt University.

Research Support

Total research funding since June 1995: $38,313,128

- Sole PI: $11,930,403
- Co-PI: $26,382,725

Grants and Contracts Received

1. "IMMortals," DARPA (through subcontract with Raytheon), 12/1/15 to 12/1/18, $1,235,567, Co-PI Jules White.
2. "The Robust Software Modeling Tool (RSMT)," ONR, 7/1/14 to 6/30/17, $749,904, Co-PI Jules White.
3. "Building Resilient Distributed Systems for Next Generation Mobile Adhoc Cyber Physical Systems," Siemens 9/1/14 to 8/31/15, $238,188.
4. "Industrial Internet Architecture," Varian Medical Systems, Inc., 10/1/14 to 10/30/15, $69,404, co-PI Jules White
5. "Capability-Based Technical Reference Frameworks for Open System Architecture Implementations," OSD ASDR&E, 7/3/14 to 9/11/14, $29,690.
6. "Progressive Model Generation for Adaptive Resilient System Software," ONR STTR, 8/6/13 to 1/31/14, $49,406, co-PI Jules White.
7. "Systems and Software PRodUcibility Collaboration and Experimentation Environment (S2PRUCE2)," AFRL (subcontract through Lockheed Martin Advanced Technology Lab), 1/4/13 to 9/30/13, $108,645, with A. Gokhale.
8. "Stochastic Hybrid Systems Modeling and Middleware-enabled DDDAS for Next-generation US Air Force Systems," AFOSR, 10/1/13 to 9/30/16, $935,402, Co-PI(s) Aniruddha Gokhale and Xenofon Koutsoukous.
9. "Workshop on Computing Clouds for Cyber Physical Systems," NSF, 9/15/12 to 12/31/2013, $73,738.
10. "Using Social Learning to Improve Adolescent Diabetes Protocol Adherence," NIH, $1,798,029, 9/1/12-8/31/16, PI Shelagh Mulvaney.

11. "Systems and Software PRodUcibility Collaboration and Experimentation Environment (S2PRUCE2)," AFRL (subcontract through Lockheed Martin Advanced Technology Lab), 4/3/08 to 9/30/12, $381,708, with A. Gokhale.
12. "Team for Research in Ubiquitous Secure Technology (TRUST)," NSF (subcontract through UC Berkeley), 6/1/05 to 10/31/15, $5,970,900, co-PI(s) J. Sztipanovits and G. Karsai.
13. "Android Mobile Military Middleware Objects (AMMO)," DARPA, 9/30/10 to 5/02/12, $1,074,093, with S. Neema.
14. "Cyber-physical multi-Core Optimization for Resource and cachE effectS (C2ORES)", AFRL, 8/1/12 to 7/31/13, $300,000, with A. Gokhale.
15. "Model-Driven Tools for Distributed- and Multi-Core Middleware," AFRL, 4/10/12 to 10/2/12, $30,000, with A. Gokhale.
16. "Cloud Environmental Analysis and Relief," NSF, 8/1/10 to 7/31/12, $66,000, with A. Gokhale.
17. "Environment-Specific Inter-ORB Protocols," SAIC, 8/1/09 to 5/23/12, $348,350, with A. Gokhale.
18. "CoSMIC and CIAO Enhancements," Northrop Grumman, 7/1/09 to 9/30/10, $878,661
19. "Integrating DDS and CCM," Northrop Grumman, 7/1/09 to 2/15/10, $85,000
20. "Early Integration and Performance Testing of Heterogeneous Computing Environments," Australian Defence Science and Technology Organization (DSTO), 1/9/09 to 7/30/09, $180,000.
21. "Predictive Cache Modeling and Analysis," AFRL (subcontract through Lockheed Martin Aeronautics), 3/1/10 to 9/30/11, $100,000.
22. "Applications of Reliable, Fast Event Notification," Raytheon, 6/1/2008 to 5/30/2009, $60,000.
23. "Open Modular Embedded Architectures," General Electric Global Research, 8/1/2008 to 1/31/2009, $35,000.
24. "Analysis and Simulation Techniques for Next-generation Motion Control Systems," Aagard, 8/1/2008 to 1/31/2009, $13,850 with Akos Ledeczi.
25. "Open Modular Embedded Architectures," Raytheon, 8/1/2008 to 3/31/2009, $74,276.
26. "NAOMI," LMCO Advanced Technology Lab, 9/1/2007 to 11/30/2009, $290,000.
27. "IU/CRC Membership," Siemens, 1/1/2009 to 12/31/2009, $40,000.
28. "Enterprise Application Configuration in the Context of Model Driven Software Development and Software Factories," Siemens Corporate Research, 10/1/07 to 9/31/08 $91,798.
29. "Modular Extendable Demonstration of an Upgradeable Space Architecture (MEDUSA)," DARPA (subcontract through Lockheed Martin Advanced Technology Center), 2/1/2008 to 1/31/2011, $600,000.
30. "CCM Middleware Implementation and Integration," PrismTech, 6/8/2007 to 3/31/2007, $33,778.

31. "The Smart Sensor Web Architecture," NASA (subcontract through Lockheed Martin Advanced Technology Center), 12/15/06 to 11/14/09, $467,728, co-PI G. Biswas.
32. "I/UCRC Membership," General motors, 1/1/2008 to 12/31/2008, $100,000, co-PI G. Karsai.
33. "Pollux: Enhanding the Real-time QoS of the Global Information Grid," AFRL, 2/24/06 to 7/24/08, $1,242,718, co-PI M. Reiter.
34. "Intelligent Middleware for Next Generation Petascale Scientific Computing," Vanderbilt Discover Grant, 5/1/05 to 6/30/07, $100,000, co-PI(s) A. Gokhale and P. Sheldon.
35. "Air Force Center for Research on GIG/NCES Challenges," AFOSR (subcontract through UC Berkeley), 3/1/06 to 2/28/08, $600,000, co-PI J. Sztipanovits.
36. "Quality of Service Enabled Dissemination," AFRL (subcontract through BBN Technologies), 12/31/2007 to 9/30/2009, $320,000.
37. "A Fault-Tolerant Real-Time CORBA Naming Service," US Navy (subcontract through Tech-X Corp), 11/1/2007 to 4/30/2010, $175,000, co-PI A. Gokhale.
38. "System Execution Modeling Technologies for Large-scale Net-centric Systems," AFRL, 1/1/2008 to 12/31/2010, $244,000.
39. "Model-Driven Computing for Distributed Real-time Embedded Systems," Raytheon, 8/31/04 to 8/31/08, $500,000.
40. "NAOMI," LMCO Advanced Technology Lab, 9/1/2007 to 11/30/2007, $50,000.
41. "ACE/TAO Improvement Techniques and Solutions, Veritas/Symantec, 3/31/05 to 4/31/08, $198,500.
42. "Adaptive Resource Control for Certificable Systems," DARPA (subcontract through LMCO Advanced Technology Lab), 3/30/2007 to 12/31/2007, $50,000.
43. "Survivable Internet-scale Distributed Systems," IDA, 3/30/2007 to 12/31/2007, $60,000.
44. "QUality of service pICKER (QUICKER)," LMCO Advanced Technology Lab, 3/30/2007 to 12/31/2007, $60,000.
45. "Thimble," LMCO Advanced Technology Lab, 3/30/2007 to 12/31/2007, $60,000.
46. "CADynCE Experimentation Operations (CEO)," DARPA (subcontract through LMCO Advanced Technology Lab), 8/31/2007 to 12/31/2007, $25,000.
47. "Real-time Discovery for Pub/Sub Middleware in WANs," US Navy (subcontract through Tech-X Corp), 6/16/2007 to 9/31/2007, $15,000.
48. "GEMS Utilization Test Suite," LMCO Advanced Technology Lab, 9/1/07 to 11/30/07, $50,000.
49. "Advanced Information Systems and Technology Program," NASA (subcontract through LMCO Advanced Technology Center), 11/13/2007 to 12/1/2007, $22,000, co-PI G. Biswas.
50. "Design for Adaptivity and Reliable Operation of Software Intensive Systems," NSF CNS-0613971, 9/1/06 to 8/31/08, $199,867, co-PI(s) S. Abdelwahed and G. Karsai.

51. "Software Technologies Targeting Interoperability for Systems of Systems," Army Research Lab, 1/15/07 1/14/10, $851,567, co-PI(s) G. Karsai and J. Sztpanovits.
52. "Software Wind Tunnel (SWiT) Capabilities," Lockheed Martin Advanced Technology Lab, 8/1/06 to 12/31/06, $60,000.
53. "High-Confidence Software Platforms for Cyber-Physical Systems," NSF, 5/1/06 to 7/30/08, $129,179.
54. "Applying AOP to Develop of Component Synthesis with MDD," Siemens, 3/1/03 to 2/28/07, $400,005.
55. "Addressing Domain Evolution Challenges in Model-Driven Software Product-lines," Siemens Corporate Research, 10/1/05 9/31/07, $100,000.
56. "A Fault Tolerant Real-time CORBA Naming Service," US Navy (subcontract through Tech-X Corp), 11/1/05 to 8/31/06, $15,000.
57. "The SYstem DEployment and Configuration AssisteR (SYDECAR)," Lockheed Martin Advanced Technology Lab, 8/1/05 to 8/1/08, $500,000.
58. "Future Combat Systems: Software Architecture Engineering," DARPA (subcontract through Boeing), 1/28/05 to 12/31/07, $2,764,226, co-PI(s) J. Sztipanovits and G. Karsai.
59. "Development of an Eclipse Plug-in," PrismTech, 4/28/05 to 9/30/05, $25,000.
60. "Prometheus: Enhancing the QoS of the JBI," AFRL, 3/25/05 to 12/31/05, $500,000, co-PI(s) K. Birman and Mike Reiter.
61. "A Testbed for Assuring Quality of Software for DRE Systems," ONR, 2/15/05 to 1/31/06, $200,000, co-PI(s) A. Gokhale and A. Porter.
62. "Enhancing the QoS of SOAs Using Eclipse-based MDD," IBM, 2/15/05 to 1/31/06, $29,515, co-PI A. Gokhale.
63. "Model-Driven Development of BEEP Application Protocols," Cisco, 12/15/04 to 12/14/05, $57,976, co-PI A. Gokhale.
64. "Evaluating CORBA Middleware for Space Systems," NASA (subcontract through Lockheed Martin Advanced Technology Center), 9/23/04 to 11/30/06, $186,180, co-PI G. Biswas.
65. "Refactoring Techniques to Reduce Middleware Resource Utilization," Qualcomm, 10/31/04 to 10/31/05, $104,000, co-P B. Natarajan.
66. "Model-Driven Development for Software Defined Radios," BAE Systems, 12/1/04 to 3/31/05, $32,000.
67. "Enhancing the Robustness and Performance of TENA," DISA (subcontract through SAIC and OSC), 7/1/04 to 12/31/04, $75,000.
68. "QoS-enabled Fault Tolerant Middleware and MDA Tools," Lockheed Martin MSS, 4/1/03 to 12/31/04, $516,434.
69. "Trustworthiness in Embedded Systems," NSF ITR CCR-032574, 9/31/03 to 8/31/06, $210,454.
70. "ACE+TAO Enhancements," OCI, gift $20,000.
71. "Acquiring Accurate Dynamic Field Data Using Lightweight Instrumentation," NSF ITR CCR-0312859, 10/1/02 to 9/31/07, $1,850,000, co-PI(s) A. Porter, D. Notkin, and A. Karr.

72. "Intergovernmental Personnel Act," DARPA, 6/1/00 to 5/31/02, $198,934.
73. "Optimizing Component Models," DARPA, 4/1/01 to 6/31/02, $210,000.
74. "HLA RTI Next-generation," DMSO (subcontract through SAIC), 6/1/01 to 12/31/01, $70,895.
75. "ACE Enhancements for Windows NT and Windows CE," Siemens Medical Engineering, 2/1/00 9/19/01, $112,000.
76. "Scalable and Fault Tolerant Middleware," AFRL MURI, 12/1/99 to 3/31/02, $253,701.
77. "Protocol Engineering Research Center," AFSOR MURI, 6/15/00 to 6/14/03, $264,720, co-PI Tatsuya Suda.
78. "Optimizing ORBs for Network Management," Cisco Systems, 1/1/00 to 12/31/00, $100,000.
79. "TAO Optimizations," Raytheon, 10/1/99 to 6/01/01, $50,000.
80. "ACE+TAO on pSoS," Motorola, 8/15/99 to 12/31/99, $30,000.
81. "Real-time Distributed Object Computing," Sprint, 8/15/99 8/14/00, $133,068.
82. "TAO Enhancements," Krones, 8/1/99 to 9/1/99, $5,000.
83. "ACE Enhancements," ICOMVERSE, gift, $20,000.
84. "Weapon Systems Open Architecture," Boeing, 7/15/99 to 1/31/00, $51,491.
85. "Fault Tolerant CORBA," Motorola Labs, 7/15/99 to 7/14/00, $139,000.
86. "TAO Enhancements," Global MAINTECH, 7/1/99 to 8/1/99, $5,000.
87. "ACE QoS Extensions," Motorola Trunking, 6/1/99 to 8/1/99, $5,000.
88. "CORBA Interceptors," Experian, 5/15/99 7/14/99, $10,000.
89. "DCOM performance evaluation," Microsoft, gift, $30,000.
90. "TAO Improvements," OCI, 4/1/99 to 9/31/00, $27,000.
91. "Middleware Optimizations," Telcordia, 2/1/99 to 1/31/00, $52,700.
92. "Minimum CORBA," Hughes Data Networking, 4/1/99 to 3/31/00, $50,000, co-PI David Levine.
93. "Framework Usage Patterns," Siemens Corporate Research, 4/1/99 to 3/31/00, $35,000.
94. "Dynamic Scheduling and Real-time ORB Optimizations," Boeing, 10/1/98 9/30/99, $184,860.
95. "Distributed Object Computing Middleware," Nortel, 11/1/98 10/31/99, $75,000.
96. "ACE subsetting," "ACE subsetting„" Nokia, 10/8/98 4/8/99, $30,000.
97. "Boeing Research Fellowship," Boeing, 9/1/98 8/31/00, $81,486.
98. "Patterns and Frameworks Reuse Curriculum," Lucent Bell Labs, 9/1/98 12/31/98, $31,200.
99. "Patterns, Frameworks, and Components," Siemens ZT, 12/1/98 5/31/00, $175,000.
100. "High availability frameworks," Lucent, 9/1/98 8/31/99, $39,400.
101. "Real-time Distributed Object Computing," Sprint, 8/1/98 7/31/99, $288,194.
102. "Distributed Object Integration for the Quorum Project," DARPA S30602-98-C-0187 (subcontract through BBN), 9/1/98 8/31/01, $448,643, co-PI(s) R. Schantz and J. Loyall.

103. "Evaluating a Framework for Dynamic Distributed Real-Time Scheduling„" USENIX, gift, $18,000.
104. "Distributed Object Computing," Microsoft, gift, $20,000.
105. "Distributed Object Visualization Environment," Lockheed Martin, 5/1/98 to 11/31/99, $54,000.
106. "Distributed Object Computing with Adaptive End-to-end QoS Guarantees," DARPA 9701561, 8/1/97 to 7/31/00, $873,625.
107. "Real-time CORBA for Telecommunications," Lucent, 12/1/97 to 11/31/98, $100,000.
108. "Developing an HLA-compliant RTI with ACE," SAIC, 12/15/97 to 1/31/00, $228,075.
109. "Real-time CORBA for Wireless," Motorola LMPS, 10/15/97 to 10/14/98, $200,000.
110. "Real-time CORBA for Avionics," Computing Devices International, 10/15/97 to 10/14/98, $39,050.
111. "Dynamic Scheduling of Real-time OFPs," Boeing, 9/1/97 to 8/31/98, $224,604.
112. "Distributed Object Visualization," Siemens MED, 10/1/97 to 9/1/98, $40,000.
113. "The ADAPTIVE Communication Environment," Siemens MED, 10/1/97 to 9/1/98, $70,000.
114. "The Architect's Assistant," Siemens Corporate Research, 9/1/97 to 8/1/98, $35,000.
115. "Monitoring, Visualization, and Control of High Speed Networks," NSF NCR-97-14698, 9/1/97 to 8/31/01, $1,200,000, co-PI(s) G. Parulkar, E. Kraemer, J. Turner, and R. Cytron .
116. "Adaptive Software Technology Demonstration (ASTD)," AFRL (subcontract through Boeing), 9/1/98 to 8/31/02, $1,200,000, co-PI(s) B. Doerr, D. Allen, and R. Jha.
117. "Patterns, Frameworks, and Components for Multimedia Systems," Siemens Research, 1/97 to 6/98, $150,000.
118. "Adaptive Servers for High-Performance Imaging," Kodak Networked Imaging Tech. Center, 11/96 to 11/97, $40,000.
119. "Real-time CORBA," Sprint, 9/96 to 12/97, $345,000, co-PI G. Parulkar.
120. "OpenMAP – Object-Oriented Components for Real-time Avionics," McDonnell Douglas, 9/96 to 9/97, $241,591.
121. "Compilation and Automatic Optimization of Network Protocol Implementations," NSF NCR-9628218, 8/96 to 8/99, $411,025, co-PI(s) G. Varghese and R. Cytron (PI).
122. "Medical Imaging with Java and the WWW," SIEMENS Medical Engineering, 8/96 to 7/97, $125,000.
123. "The ADAPTIVE Communication Environment," SIEMENS Medical Engineering, 8/96 to 7/97, $90,000.
124. "High-performance Distributed Medical Imaging," Kodak Imaging, 12/94 to 8/96, $55,152, co-PI J. Blaine.

125. "Design Patterns for Concurrent Object-Oriented Networking," Object Technologies International, 4/96 to 4/97, $25,000.
126. "Distributed Object Computing with CORBA and DCE," Bellcore, 5/96 to 12/96, $32,978.
127. "The ADAPTIVE Communication Environment," SIEMENS Medical Engineering, 6/95 to 6/96, $170,000.

Courses Taught

Courses at Vanderbilt University

9. CS 215 – Intermediate Software Design, Spring 2006
10. CS 251 – Intermediate Software Design, Spring 2007, Spring 2008, Spring 2009, Fall 2009, Spring 2010, Spring 2012, Spring 2014, Spring 2015, Spring 2016
11. CS 291/242 – Software Design Studio, Fall 2004
12. CS 291/242 – Software Design Studio, Fall 2003
13. CS 292 – Beyond the Oneway Web, Fall 2008
14. CS 278 – Software Engineering, Fall 2008
15. CS 279 – Software Engineering Projects, Spring 2010
16. CS 282 – Principles of Operating Systems II, Spring 2003, Spring 2004, Fall 2005, Fall 2007, Fall 2012, Fall 2013, Fall 2014, Fall 2015
17. CS 395 – Advanced Network Software Design, Fall 2006
18. CS 395 – QoS-enabled Middleware, Fall 2008
19. CS 396 – QoS-enabled Component Middleware, Spring 2005

Courses at Coursera

1. Pattern-Oriented Software Architecture: Concurrency, 2014, 2015, and 2016
2. Pattern-Oriented Software Architecture: Communication, 2014, 2015, and 2016
3. Pattern-Oriented Software Architecture: Content, 2016
4. Pattern-Oriented Software Architectures for Concurrent and Networked Software, 2013
5. Pattern-Oriented Software Architecture for Concurrent and Networked Software, 2012

Courses at University of California, Irvine

1. ECE 011 – Computational Methods in ECE, Winter 2000
2. ECE 255 – Distributed Software Architecture Design, Spring 2000
3. ICS 142 – Compiler Theory, Summer 1989
4. ICS 23 – Data Structures, Summer 1988

Courses at Washington University, St. Louis

1. CS 562 – Advanced Object-Oriented Software Development with Patterns and Frameworks, Spring 1999
2. CS 242 – Introduction to Software Design, Spring 1998
3. CS 673 – Distributed Systems research seminar, Fall 1997
4. CS 422 – Operating Systems Organization, Fall 1997
5. CS 242 – Introduction to Software Design, Spring 1997
6. CS 544 – Distributed System Design, Fall 1996
7. Ada tasking course for McDonnell Douglas, Fall 1996
8. OO design course for McDonnell Douglas, Spring 1996
9. CS 523 – Distributed Operating Systems Organization, Spring 1995
10. CS 242 – Introduction to Software Design, Fall 1995
11. CS 673 – Distributed Systems research seminar, Spring 1995
12. CS 422 – Operating Systems Organization, Fall 1994

Other Teaching Experience

- In addition to the academic teaching experience above, I have also taught numerous short-courses and tutorials on object-oriented design patterns and programming techniques, UNIX and Windows NT systems programming and network programming, C++ and C programming languages, and various distributed operating system, compiler construction, algorithm, and data structure courses for the following universities and professional organizations:
  1. Pearson LiveLessons
  2. University Extension Program, University of California, Berkeley, CA
  3. University Extension Program, University of California, Irvine, CA
  4. University Extension Program, University of California, Los Angeles, CA
  5. Oregon Graduate Institute of Science and Technology, Beaverton, OR
  6. USENIX association
  7. Association of Computing Machinery (ACM)
  8. Addison-Wesley's Technology Exchange Program, Reading, MA
  9. SIGS Conferences
  10. Object Computing Institute, St. Louis, MO
  11. National University, Irvine, CA

Department/School/Community Service

Service at Vanderbilt University

1. Member of the search committee for the first Director of the Innovation Center

1. VUSE point of contact for VUIT

2. Member of the VUIT faculty advisory committee

3. Owen-VUSE joint committee for 2014-2015

4. Chair of the Schmidt Family Annual Educational Technologies Lectureship

5. Committee member for Bobby Bodenheimer's promotion case to full professor
6. Committee member for Julie Adams's promotion case to full professor
7. Committee member for Akos Ledeczi's promotion case to full professor
8. Member of the Provost's Study Group on Cross College Teaching
9. Member of the Advisory Committee for the Vanderbilt Institute for Digital Learning (VIDL)
10. Chair of the Provost's Committee on the Innovation Center
11. Chair of two year committee for Eugene VorobeychikMember of the Chancellor's Social Media and the Internet committee
12. Faculty advisor for the Vandy Mobile student organization
13. Chair of the VUSE Technology Entrepreneurship Task Force
14. Member of the VU Online Education Task Force
15. Chair of the CS search committee
16. Member of the VUSE Career Committee
17. Member of the Advisory Group to guide mobile app development at Vanderbilt
18. Chair of the tenure committee for Yuan Xue
19. Chair of the four year review committee for Yuan Xue
20. Member of the two year committee for Yuan Xue
21. Member of the ad hoc committee on EECS Industrial Advisory Board
22. Ex-officio member of the ad hoc committee on the CS graduate program
23. Ex-officio member of the ad hoc committee on the CS undergraduate program
24. Member of the promotion committee for Ted Bapty
25. Faculty facilitator for the Vanderbilt Visions program
26. Chair of the Information Technology committee for the Vanderbilt School of Engineering
27. Chair of the tenure committee for Bobby Bodenheimer

28. EECS Corporate/Internship Liaison for Computer Science and Engineering
29. Member of review committee for Xenofon Koutsoukos
30. Ex-officio Member of the Ad Hoc Committee on Computer Engineering
31. Faculty sponsor of the new EECS Graduate Student Organization
32. Member of the VUSE Research Institutes and Centers Council
33. Associate Chair of Computer Science and Engineering
34. Member of the Vanderbilt University Faculty Senate
35. Chair of promotion committee for Gabor Karsai
36. Chair of 2005 faculty recruiting committee
37. Member of promotion committee for Gautam Biswas
38. Chair of the faculty committee on Academic Computing and Information Technology (ACIT)
39. Member of the Research Advisory Committee on Information Technology (RACIT)
40. Chair of the Systems Engineering concentration committee
41. Member of the Plan Integration and Communication Group (PICG)
42. Chair of 2003 faculty recruiting committee
43. Member of the CS graduate curriculum committee

Service at Washington University, St. Louis

1. Member of the Faculty recruiting committee
2. Member of the CS committee on recruiting industrial graduate students (RIGS)
3. Member of the CS Experimental Infrastructure for Teaching and Research (CEITR)
4. Member of the Introductory course committee
5. Member of the Graduate admission committee
6. Member of the CS representative to the CEC advisory board
7. Member of CS departmental chair search committee

Awards and Honors

1. Received the 2015 Award for Excellence in Teaching by the Vanderbilt University School of Engineering.

2. Vice-chair of the IEEE Chapter in middle Tennessee.
3. Elected to three year term as member of the Vanderbilt University Faculty Senate.
4. Invited speaker at the dedication of the Henry Samueli School of Engineering, along with UC Irvine Chancellor, Ralph Cicerone; Dean of the School of Engineering, Nicolaos Alexopoulos; Chairperson of the Regents of the University of California, S. Sue Johnson; President of the University of California, Dick Atkinson; and CTO and co-founder of Broadcom Henry Samueli.
5. Received early promotion to tenure as an Associated Professor at Washington University, St. Louis, five years after joining the faculty as an Assistant Professor in 1994.
6. Director of the "Center for Distributed Object Computing" at Washington University, St. Louis since spring of 1999.
7. Received joint appointment to the Mallinckrodt Institute Department of Radiology, Washington University School of Medicine, February 1996.
8. Invited by Dr. Martina Zitterbart to participate in a 4-week international exchange program at the Universität Karlsruhe Institut für Telematik in Karlsruhe, Germany, April 1993.

Consulting Work

1. ARINC, Fountain Valley, CA
2. ACM, NY, NY
3. Advanced Institute of Information Technology, Seoul, Korea
4. AG Communication Systems, Phoenix, AZ
5. Anderson Consulting, Chicago, IL
6. Apple, Cupertino, CA
7. AT&T Research, Murray Hill, NJ
8. BAE Systems, Greenlawn, NY
9. BAE Systems, Wayne, NJ
10. BEA, San Jose, CA
11. Bellcore, Morristown, NJ
12. BellSouth, Atlanta, GA
13. Boeing, St. Louis, MO
14. Boies, Schiller, and Flexner, Santa Monica, CA
15. Bridges and Mavrakakis, Palo Alto, CA
16. Credit Suisse, Zurich, Switzerland
17. Crosskeys, Ottawa, Canada
18. DARPA, Arlington, VA
19. Edward D. Jones, St. Louis, MO
20. Envision Inc. St. Louis, MO
21. Ericsson, Cypress, CA
22. GaN Corporation, Huntsville, AL
23. Gibson, Dunn, and Crutcher, NY, NY
24. Jet Propulsion Lab, Pasadena, CA
25. Keystone Strategy, Boston, MA

26. Kilpatrick Stockton, Atlanta, GA
27. Kirkland and Ellis, San Francisco, CA
28. Kodak Imaging, Rochester, NY
29. Laureate University, Baltimore, MD
30. Lockheed Martin Tactical Systems, Minneapolis, MN
31. Lockheed Martin Mission Systems, Boulder, CO
32. Lockheed Martin Advanced Technology Lab, Cherry Hill, NJ
33. Lucent Bell Labs, Naperville, IL
34. Lucent Bell Labs, Murray Hill, NJ
35. Lucent, Whippany, NJ
36. McDonnell Douglas, St. Louis, MO
37. Microsoft, Redmond, WA
38. Morrison and Foerster, Washington DC
39. Morgan Stanley, New York, NY
40. Motorola Cellular Infrastructure Group, Arlington Heights, IL
41. Motorola Iridium, Chandler, AZ
42. Motorola Land Mobile Products, Chicago, IL
43. National Security Agency, Ft. Meade, MD
44. Naval Air Weapons Stations, China Lake, CA
45. Nortel, Ottawa, Canada
46. Object Computing Institute, St. Louis, MO
47. Object Technologies International, Ottawa, CA
48. Odetics Broadcasting, Anaheim, CA
49. Oracle, Redwood Shores, CA
50. Park, Vaughan, and Fleming, Boise, ID
51. Pragmatus, Alexandria VA
52. PrismTechnologies, Newcastle, UK
53. Qualcomm, San Diego, CA
54. Raytheon, San Diego, CA
55. Riverace, Boston, MA
56. SAIC, Washington D.C.
57. Schwegman, Lundbert, and Woessner, Minneapolis, MN
58. Siemens Medical Engineering, Erlangen, Germany
59. Siemens Corporate Research, Princeton, NJ
60. SIGS, New York, NY
61. Software Engineering Institute, Pittsburgh, PA
62. Teradyne, Chicago, IL
63. UC Berkeley Extension, Palo Alto, CA
64. UCLA Extension, Los Angeles, CA
65. USENIX, Lake Forest, CA
66. Wong, Cabello, Lutsch, Rutherford & Brucculeri, Houston, TX
67. WMS Gaming, Chicago, IL
68. Zircon Computing, Wayne, NJ

Expert Testimony in the Past Five Years

1. Declaration of Douglas C. Schmidt in support of plaintiff's motion for reconsideration of the Court's order at DKT. No 231, United States District Court for the District of Idaho, Hoyt A. Fleming, Plaintiff, v. Escort Inc, Defendants. Case No. 1:12-cv-0066-BLW

Summary of Research Contributions

Over the past three decades, I have led influential R&D efforts at the University of California, Irvine; Washington University, St. Louis; the Defense Advanced Projects Research Agency (DARPA); Vanderbilt University, and Carnegie Mellon University. In these efforts, I have conducted and managed research projects on a range of topics, including patterns, optimization techniques, and empirical analyses of software frameworks that facilitate the development of quality of service (QoS)-enabled middleware and model-driven engineering (MDE) techniques/tools for distributed real-time and embedded (DRE) systems and mobile cloud computing apps running over wired/wireless networks and embedded system interconnects.

At Vanderbilt University I direct the Distributed Object Computing (DOC) Group at the Institute for Software Integrated Systems (ISIS), which is one of the leading research groups on middleware platforms and MDE tools for DRE systems and mobile cloud computing platforms. The R&D efforts I have led have had a significant impact on academic research and commercial practice. My work has been cited more than 25,000 times across a comprehensive spectrum of high-impact publications and my h-index is 74, which indicates the significant impact of my publications as a researcher in the field of Computer Science.

Scores of universities throughout the world also use the middleware and MDE tools my DOC Group has developed as the basis for their research and teaching fforts. Moreover, the middleware platforms and MDE tools developed by my DOC Group is used by developers in thousands of companies (such as Boeing, Cisco, Ericsson, Kodak, Lockheed Martin, Lucent, Motorola, NASA/JPL, Nokia, Nortel, Raytheon, SAIC, Siemens, Sprint, and Telcordia) in a wide range of domains (such as telecom/datacom, healthcare, process automation, avionics, homeland security and defense, financial services, online gaming, and distributed interactive simulation).

The remainder of this section summarizes my career accomplishments and describes the key contributions of my research and cites representative examples of my publications.

1) Summary of Career Accomplishments

My career accomplishments include the following:

Publications and presentations. I have published over 580 works (112 journal papers, 180 conference papers, 5 books, 4 book-length reports, 3 edited book collections, 64 book chapters, 70 workshop papers, 13 short papers and posters, 73

trade magazine columns/articles, and 63 editorials and book forewords). My papers have appeared in the most selective journals (*e.g.*, ACM Transactions in Embedded Computing Systems, IEEE Transactions on Parallel and Distributed Systems, IEEE Transactions on Software Engineering, IEEE Transactions on Computing, IEEE Journal of Selected Areas of Communications, and ACM Transactions on Autonomous and Adaptive Systems) and conferences (*e.g.*, ACM SIGCOMM, ACM OOPSLA, IEEE INFOCOM, IEEE ICDCS, IEEE RTAS, ACM/IEEE Middleware, and the ACM/IEEE ICSE) in my field. I have also given over 500 invited lectures and tutorials world-wide.

Funding. Since June 1995 I have been a PI or co-PI for grants, contracts, and gifts totaling more than $38 million dollars. I have been the sole PI for over $11.5 million dollars of this amount.

Graduate advising and training. During my academic career at Washington University, UC Irvine, and Vanderbilt University, I have graduated over 25 masters students and 18 doctoral students.

Professional service. I have engaged in the following professional service capacities:

- Served as guest editor of 12 ACM, IEEE, and USENIX journals, and served as editor-in-chief and associate editor of the C++ Report magazine.
- Served as general chair or program (co)-chair for 35 conferences, tutorial chair for 4 conferences, co-organized 14 workshops, and served on the program committees for over 245 IEEE, ACM, IFIP, USENIX, and OMG conferences.
- Served a member of the Air Force Scientific Advisory Board, where I was the Vice Chair of a study on Cyber Situational Awareness for Air Force mission operations.
- Serve on the Advisory Board for the Future Airborne Capability Environment (FACE)
- Served as co-lead of a task area on "Published Open Interfaces and Standards" for the US Navy's Open Systems Architecture initiative.
- Served as a Program Manager at the DARPA Information Technology Office (ITO) and Information eXploitation Office (IXO) the Deputy Director for DARPA ITO, helping to set the national agenda on IT research and development for the USA.
- Served as Co-chair for the Software Design and Productivity (SDP) Coordinating Group, which formulates the multi-agency research agenda in fundamental software design for the Federal government's Information Technology Research and Development (IT R&D) Program, which is the collaborative IT research effort of the major Federal science and technology agencies.

2) Description of Research Contributions

My research on middleware and model driven engineering (MDE) has produced concurrent and networked software frameworks, services, protocols, and tools that

enable DRE systems and mobile cloud computing apps to invoke operations on objects without concern for their location, programming language platform, operating system, and/or hardware. Software for these types of applications must be *flexible*, *efficient*, and *predictable*. Flexibility is necessary to respond rapidly to application requirements that span an increasingly wide range of media types and traffic patterns. Efficiency and predictability are necessary to support the QoS demands of performance-sensitive and time-sensitive applications.

Despite dramatic increases in the performance of networks and computers, designing and implementing flexible, efficient, and predictable DRE systems and mobile cloud computing apps remains hard, and substantial time and effort is required to develop and deploy these systems and apps today. My research has therefore focused on innovative techniques, patterns, and MDE tools that have improved DRE system and mobile cloud computing app development by:

- Identifying and alleviating key performance bottlenecks and sources of priority inversion and non-determinism in middleware platforms over local-area and wide-area wireless/wired networks.
- Discovering and formalizing patterns and pattern languages to enhance the development and evolution of middleware and MDE tools to meet the stringent QoS requirements of DRE systems and mobile cloud computing apps.
- Developing innovative optimization techniques, middleware frameworks and components, and MDE tools that (1) simplify and automate middleware software development, validation, and evolution and (2) achieve high-performance, low latency, and real-time predictability end-to-end across wireless/wired networks and embedded interconnects.

My specific research areas and contributions are described below. Publications related to my DOC Group research activities are available at www.dre.vanderbilt.edu/~schmidt.

a) Systematically Measuring the Performance of Middleware Over High-speed Networks and Embedded Systems Interconnects

My DOC Group has developed and employed middleware testbed environments to conduct extensive experiments that systematically identify the performance bottlenecks and sources of priority inversion and non-determinism in communication middleware software on high-speed networks. The experiments in our testbed have studied *lower-level network programming mechanisms*, such as socket-based C interfaces and the C++ wrappers for sockets, and *higher-level middleware*, such as Real-time CORBA, which is an open international standard for distributed object computing that has been highly influenced by the our R&D on patterns and middleware.

Our experiments on middleware performance have received widespread recognition in academia and industry. As a direct result of the analysis in our work, for instance, many CORBA suppliers have tuned their ORB implementations to improve performance considerably. The current generation of Real-time CORBA ORBs and Data Distribution

System (DDS) middleware are now competitive with hand-coded C/C++ TCP/IP implementations. This improvement is important for performance-sensitive, mission-critical DRE application domains, such as real-time avionics and shipboard computing, where the use of higher-level middleware greatly decreases development effort and increases system reliability and flexibility.

Papers published in top journals and conferences related to our empirical studies of middleware performance include:

1. Ming Xiong, Jeff Parsons, James Edmondson, and Douglas C. Schmidt, "Evaluating Technologies for Tactical Information Management in Net-Centric Systems, Proceedings of the Defense Transformation and Net-Centric Systems conference, April 9-13, 2007, Orlando, Florida.
2. Arvind S. Krishna, Nanbor Wang, Balachandran Natarajan, Aniruddha Gokhale, Douglas C. Schmidt and Gautam Thaker, "CCMPerf: A Benchmarking Tool for CORBA Component Model Implementations", *International Journal of Time-Critical Computing Systems*, Springer, Vol. 29, No. 2-3, pp. 281-308, March-April 2005.
3. Angelo Corsaro and Douglas C. Schmidt, "The Design and Performance of Real-time Java Middleware," Special Issue on Middleware for the *IEEE Transactions on Parallel and Distributed Systems*, Volume 14, Number 11, November 2003.
4. Andy Gokhale and Douglas C. Schmidt, "Measuring and Optimizing CORBA Latency and Scalability Over High-speed Networks," *IEEE Transactions on Computing*, April, 1998.
5. Aniruddha Gokhale and Douglas C. Schmidt, "Measuring the Performance of Communication Middleware on High-Speed Networks," Proceedings of SIGCOMM '96, ACM, San Francisco, August 28-30th, 1996.

b) Developing Innovative Middleware Optimization Techniques

Based on the result of the empirical performance studies described above, my DOC Group has developed middleware optimization techniques that yield highly efficient and predictable Object Request Broker (ORB) implementations, without sacrificing flexibility, reuse, or standards-conformance. We have applied and demonstrated these optimization techniques in the context of:

- **ADAPTIVE Communication Environment** (ACE), which is an object-oriented toolkit containing frameworks and components that implement key patterns for DRE systems.
- **The ACE ORB** (TAO), which is a high-performance, real-time ORB targeted for DRE systems with hard and soft QoS requirements.
- **The Component-Integrated ACE ORB** (CIAO), which is a real-time CORBA Component Model (CCM) implementation built on top of TAO.

ACE, TAO, and CIAO are open-source software that have been used in thousands of DRE systems around the world. As a testament to our success in technology transfer, many companies (including OCI, PrismTechnologies, Remedy, and Riverace) provide commercial support for ACE, TAO, and CIAO using an open-source business model.

The following is a synopsis of the key research contributions and publications stemming from the ACE, TAO, and CIAO projects:

- A **real-time ORB Core** that supports deterministic scheduling and dispatching strategies. TAO's and CIAO's ORB Core concurrency models are designed to minimize context switching, synchronization, dynamic memory allocation, and data movement. TAO and CIAO were the first standards-based ORBs with these capabilities. Representative papers published on this topic include:

  1. Venkita Subramonian, Gan Deng, Christopher Gill, Jaiganesh Balasubramanian, Liang-Jui Shen, William Otte, Douglas C. Schmidt, Aniruddha Gokhale, and Nanbor Wang, "The Design and Performance of Component Middleware for QoS-enabled Deployment and Configuration of DRE Systems," *Elsevier Journal of Systems and Software*, Special Issue Component-Based Software Engineering of Trustworthy Embedded Systems, volume 80, number 5, March, 2007.
  2. Douglas C. Schmidt, Sumedh Mungee, Sergio Flores-Gaitan, and Aniruddha Gokhale, "Software Architectures for Reducing Priority Inversion and Non-determinism in Real-time Object Request Brokers," *Journal of Real-time Systems*, Kluwer, Vol. 21, No. 2, 2001.
  3. Douglas C. Schmidt, "Evaluating Architectures for Multi-threaded CORBA Object Request Brokers," *Communications of the ACM*, Special Issue on CORBA, ACM, edited by Krishnan Seetharaman, Volume 41, No. 10, October 1998.

- An **optimal active demultiplexing strategy** that associates client requests with target objects in constant time, regardless of the number of objects and operations. TAO was also the first ORB with these capabilities. Representative papers published on this topic include:

  9. Irfan Pyarali, Carlos O'Ryan, Douglas C. Schmidt, Nanbor Wang, Vishal Kachroo, and Aniruddha Gokhale, "Using Principle Patterns to Optimize Real-time ORBs," *IEEE Concurrency*, Volumn 8, Number 1, January-March 2000.
  10. Andy Gokhale and Douglas C. Schmidt, "Measuring and Optimizing CORBA Latency and Scalability Over High-speed Networks," *IEEE Transactions on Computing*, April, 1998.

- A **highly optimized CORBA IIOP protocol engine** and a **highly optimizing IDL compiler** that generates compiled and/or interpreted stubs and skeletons, which enables applications to make fine-grained time/space tradeoffs. Representative papers published on this topic include:

  1. Alexander B. Arulanthu, Carlos O'Ryan, Douglas C. Schmidt, Michael Kircher, and Jeff Parsons, "The Design and Performance of a Scalable ORB Architecture for CORBA Asynchronous Messaging," Proceedings of the IFIP/ACM Middleware 2000 Conference, Pallisades, New York, April 3-7, 2000.

2. Andy Gokhale and Douglas C. Schmidt, "Optimizing a CORBA IIOP Protocol Engine for Minimal Footprint Multimedia Systems," *IEEE Journal on Selected Areas in Communications*, September, 1999.
3. Andy Gokhale and Douglas C. Schmidt, "Techniques for Optimizing CORBA Middleware for Distributed Embedded Systems" Proceedings of INFOCOM '99, March 21-25th, New York, New York.

- A **real-time I/O subsystem** that minimizes priority inversion interrupt overhead over high-speed ATM networks and real-time interconnects, such as VME. Representative papers published on this topic include:

1. Fred Kuhns, Douglas C. Schmidt, Carlos O'Ryan, and David L. Levine, "Supporting High-performance I/O in QoS-enabled ORB Middleware," *Cluster Computing: the Journal on Networks, Software, and Applications*, Volume 3, Number 3, 2000.
2. Fred Kuhns, Douglas C. Schmidt, David Levine, and Rajeev Bector, "The Design and Performance of a Real-time I/O Subsystem," Proceedings of the 5th IEEE Real-Time Technology and Applications Symposium (RTAS99), Vancouver, British Columbia, Canada, June 2-4, 1999.

- **Real-time event and scheduling services** that integrate the capabilities of TAO described above to form the basis for next-generation DRE systems for many research and commercial projects, including Boeing, Cisco, Lockheed Martin, Raytheon, Siemens, and SAIC. Representative papers published on this topic:

1. Akram Hakiri, Pascal Berthoua, Aniruddha Gokhale, Douglas C. Schmidt, Gayraud Thierry, "Supporting SIP-based Data Distribution Service End-to-End QoS in WANs," the Elsevier Journal of Systems and Software, Volume 95, September 2014, pp. 100-121.
2. Kyoungho An, Aniruddha Gokhale, Sumant Tambe, Gerardo Pardo-Castellote, and Douglas C. Schmidt, "Content-based Filtering Discovery Protocol (CFDP): Scalable and Efficient OMG DDS Discovery Protocol," 8th ACM International Conference on Distributed EventBased Systems, Mumbai, India, May 26-29, 2014.
3. Akram Hakiri, Pascal Berthoua, Aniruddha Gokhale, Douglas C. Schmidt, Gayraud Thierry, "Supporting End-to-end Scalability and Real-time Event Dissemination in the OMG Data Distribution Service over Wide Area Networks," Elsevier Journal of Systems and Software, volume 86, number 10, October, 2013, pp. 2574–2593.
4. Irfan Pyarali, Douglas C. Schmidt, and Ron Cytron, "Techniques for Enhancing Real-time CORBA Quality of Service," the *IEEE Proceedings* Special Issue on Real-time Systems, co-editors Yann-Hang Lee and C. M. Krishna, Volume 91, Number 7, July 2003.
5. Christopher D. Gill, Douglas C. Schmidt, and Ron Cytron, "Multi-Paradigm Scheduling for Distributed Real-Time Embedded Computing," *IEEE*

Proceedings Special Issue on Modeling and Design of Embedded Systems, Volume 91, Number 1, January, 2003.

6. Chris Gill, David Levine, and Douglas C. Schmidt, "The Design and Performance of a Real-Time CORBA Scheduling Service," *International Journal of Time-Critical Computing Systems*, special issue on Real-Time Middleware, guest editor Wei Zhao, Volume 20, Number 2, March 2001.
7. Tim Harrison and David Levine and Douglas C. Schmidt, "The Design and Performance of a Real-time CORBA Event Service," Proceedings of OOPSLA '97, ACM, Atlanta, GA, October 1997.

- **Dynamic resource management techniques** that perform allocation of resources (such as memory, computational power, and network bandwidth) as well as manage application QoS and system resource utilization in various operating conditions by performing necessary control actions for many research and commercial projects, including Lockheed Martin, Raytheon, and Siemens. Representative papers published on this topic include:

1. Hamilton Turner, Brian Dougherty, Jules White, Jonathan Preston, Russell Kegley, Douglas C. Schmidt, and Aniruddha Gokhale, "DRE System Performance Optimization with the SMACK Cache Efficiency Metric," Elsevier Journal of Systems and Software, Volume 98, 2014, pp. 25-43.
2. Jules White, Brian Dougherty, Chris Thompson, Douglas C. Schmidt, "ScatterD: Spatial Deployment Optimization with Hybrid Heuristic/Evolutionary Algorithms," ACM Transactions on Autonomous and Adaptive Systems Special Issue on Spatial Computing, Volume 6 Issue 3, September 2011, 18:1–18:25.
3. Nishanth Shankaran, John Kinnebrew, Xenofon Koutsoukos, Chenyang Lu, Douglas C. Schmidt, and Gautam Biswas, "An Integrated Planning and Adaptive Resource Management Architecture for Distributed Real-time Embedded Systems," IEEE Transactions on Computers, Special Issue on Autonomic Network Computing, Special Issue on Autonomic Network Computing, volume 58, number 11, pp. 1485-1498, November 2009.
4. Nishanth Shankaran, Douglas C. Schmidt, Xenofon D. Koutsoukos, Yingming Chen, and Chenyang Lu, "Design and Performance Evaluation of an Adaptive Resource Management Framework for Distributed Real-time and Embedded Systems," EURASIP Journal on Embedded Systems (EURASIP JES): Special issue on Operating System Support for Embedded Real-Time Applications, Edited by Alfons Crespo, Ismael Ripoll, Michael Gonzalez Harbour, and Giuseppe Lipari, 2008, Pgs. 47-66.
5. Nishanth Shankaran, Xenofon Koutsoukos, Chenyang Lu, Douglas C. Schmidt, and Yuan Xue, "Hierarchical Control of Multiple Resources in Distributed Real-time and Embedded Systems," the Springer Real-time Systems Journal, Volume 39, Numbers 1-3, August, 2008, pgs. 237-282.
6. Patrick Lardieri, Jaiganesh Balasubramanian, Douglas C. Schmidt, Gautam Thaker, Aniruddha Gokhale, and Tom Damiano, "A Multi-layered Resource

Management Framework for Dynamic Resource Management in Enterprise DRE Systems," *Journal of Systems and Software*, vol. 80, Issue 7, July 2007.

7. Richard E. Schantz, Douglas C. Schmidt, Joseph P. Loyall, and Craig Rodrigues, Controlling Quality-of-Service in Distributed Real-time and Embedded Systems via Adaptive Middleware, the Wiley *Software Practice and Experience* journal, vol. 36, no. 11-12, September 2006.

c) Patterns for Developing and Evolving Middleware for DRE Systems

Achieving widespread reuse of middleware requires a concerted focus on the core *patterns* that underlie middleware and applications. Patterns formalize design expertise and articulate time-proven solutions to forces and problems that arise when developing software. Patterns also aid the development of middleware and applications by expressing the structure and collaboration of components at a level higher than source code or software design models that focus on individual functions, objects, and classes.

During the development of ACE, TAO, and CIAO my DOC Group has identified and captured a *pattern language* of essential middleware patterns for concurrency and networked collaboration. These patterns include the *Acceptor-Connector*, *Active Object*, *Asynchronous Completion Token*, *Component Configurator*, *Double-Checked Locking Optimization*, *Extension Interface*, *Half-Sync/Half-Async*, *Interceptor*, *Leader/Followers*, *Monitor Object*, *Proactor*, *Reactor*, *Scoped Locking*, *Strategized Locking*, *Thread-Safe Interface*, *Thread-Specific Storage*, and *Wrapper Facade*. Our experience applying these patterns throughout ACE, TAO, and CIAO underscores their importance in generating flexible, efficient, and predictable middleware and application software architectures for DRE systems.

Discovering, articulating, and implementing the key patterns via ACE, TAO, and CIAO enabled us to develop middleware that can support applications with statistical, *e.g.*, multimedia applications, and deterministic, *e.g.*, avionics flight and mission control systems, QoS requirements. When these patterns are reified into reusable software frameworks and components, they yield middleware that is considerably more efficient and predictable than is possible using existing middleware technologies. In particular, patterns facilitate reuse of middleware when other forms of reuse are infeasible, *e.g.*, due to fundamental differences in operating system mechanisms or programming language features.

Representative publications related to this topic include:

1. Michael Stal, Douglas C. Schmidt, and Will Otte, "Efficiently and Transparently Automating Scalable On-demand Activation and Deactivation of Services with the Activator Pattern," Software: Practice and Experience, special issue on Pattern Languages: Addressing Challenges, Edited by Mohamed Fayad and Shivanshu Singh, volume 41, number 10, October 2011, Wiley and Sons, pp. 1-16.
2. Frank Buschmann, Kevlin Henning, and Douglas C. Schmidt, "Past, Present, and Future Trends in Software Patterns," *IEEE Software*, Vol. 24, No. 4, July/August, 2007.

3. Frank Buschmann, Kevlin Henney, and Douglas C. Schmidt, *Pattern-Oriented Software Architecture: On Patterns and Pattern Languages*, Wiley and Sons, 2007.
4. Frank Buschmann, Kevlin Henney, and Douglas C. Schmidt, *Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing*, Wiley and Sons, 2007.
5. Douglas C. Schmidt, Michael Stal, Hans Rohert, and Frank Buschmann, *Pattern-Oriented Software Architecture: Patterns for Concurrent and Networked Objects*, Wiley and Sons, 2000.

d) Creating and Validating Model-driven Engineering Techniques and Tools

Despite advances in middleware technologies, key challenges must be overcome to create and validate mission-critical DRE systems effectively and productively. For example, developers of DRE systems and mobile cloud computing apps continue to use *ad hoc* means to select and compose their software due to the lack of formally analyzable and verifiable building block components. To address these issues, my DOC Group has created model-driven development (MDE) techniques and tools can be used to specify, analyze, optimize, synthesize, validate, and deploy middleware platforms and applications that can meet the needs of mission-critical DRE systems and mobile cloud computing apps. MDE is an emerging paradigm that combines

- **Metamodeling**, which define type systems that precisely express key characteristics and constraints associated with particular DRE system application domains, such as software defined radios, avionics mission computing, and total ship computing environments.
- **Domain-specific modeling languages** (DSMLs), which provide programming notations that formalize the process of specifying application logic and quality of service (QoS)-related requirements in DRE systems.
- **Model transformations and code generation** that automate and ensure the consistency of software implementations with analysis information associated with functional and QoS requirements captured by models of DRE system structure and behavior.

Our work on MDE technologies has yielded CoSMIC and GEMS, which are an open-source toolsuites containing an integrated set of MDE tools that address key lifecycle development challenges of DRE middleware and applications, such as modeling of DRE system deployment and configuration capabilities, their QoS requirements, and QoS adaptation policies used for DRE system QoS management. The CoSMIC and GEMS MDE tools enable developers of DRE systems to specify, develop, compose, and integrate application and middleware software. Representative publications related to CoSMIC and GEMS include:

1. Brian Dougherty, Jules White, and Douglas C. Schmidt, "Model-driven Auto-scaling of Green Cloud Computing Infrastructure," the Elsevier International Journal of Future Generation Computing Systems, Special Issue on Green Computing Systems, Volume 28, Number 2, February, 2012 Pages 371-378.

2. Jules White, Harrison Strowd, and Douglas C. Schmidt, "Creating Self-healing Service Compositions with Feature Modeling and Microrebooting," the International Journal of Business Process Integration and Management (IJBPIM), Special issue on Model-Driven Service-Oriented Architectures, Inderscience Publishers, pp. 35-46, Volume 4, Number 1, 2009.

3. Jules White, Douglas C. Schmidt, and Aniruddha Gokhale, "Simplifying Autonomic Enterprise Java Bean Applications via Model-driven Engineering and Simulation," *Journal of Software and System Modeling*, Vol. 7, No. 1, February, 2008.

4. Krishnakumar Balasubramanian, Jaiganesh Balasubramanian, Jeff Parsons, Aniruddha Gokhale, and Douglas C. Schmidt, A Platform-Independent Component Modeling Language for Distributed Real-time and Embedded Systems, *Elsevier Journal of Computer and System Sciences*, Vol. 73, No. 2, March 2007.

5. Douglas C. Schmidt, "Model-Driven Engineering," *IEEE Computer*, Vol. 39. No. 2, February 2006.

3) Novel Distributed Continuous Quality Assurance Techniques

Software engineers increasingly emphasize agility and flexibility in their designs and development approaches. They increasingly use distributed development teams, rely on component assembly and deployment rather than green field code writing, rapidly evolve the system through incremental development and frequent updating, and use flexible product designs supporting extensive end-user customization.

While agility and flexibility have many benefits, they also create an enormous number of potential system configurations built from rapidly changing component implementations. Since today's quality assurance (QA) techniques do not scale to handle highly configurable systems, we have developed and validated the Skoll environment, which provides novel software distributed continuous QA processes and tools that leverage the extensive computing resources of user and developer communities in a distributed, continuous manner to improve software quality significantly. Representative publications related to Skoll and distributed continuous quality assurance as part of our R&D activities include:

James Hill, Pooja Varshneya, and Douglas C. Schmidt, "Evaluating Distributed Real-time and Embedded System Test Correctness using System Execution Traces," Central European Journal of Computer Science, Volume 1, Number 2, August 2011, pp. 167-184.

James Hill, James Edmondson, Aniruddha Gokhale, and Douglas C. Schmidt, "Tools for Continuously Evaluating Distributed System Qualities," IEEE Software, July/August, 2010, Volume 27, Number 4, pp. 65-71.

Adam Porter, Atif Memon, Cemal Yilmaz, Douglas C. Schmidt, Bala Natarajan, "Skoll: A Process and Infrastructure for Distributed Continuous Quality Assurance," *IEEE Transactions on Software Engineering*, August 2007, 33(8).

Cemal Yilmaz, Adam Porter, Arvind Krishna, Atif Memon, Douglas C. Schmidt, Aniruddha Gokhale, and Bala Natarajan, "Reliable Effects Screening: A Distributed Continuous Quality Assurance Process for Monitoring Performance Degradation in Evolving Software Systems," *IEEE Transactions on Software Engineering*, February 2007, 33(2).

Cemal Yilmaz, Atif Memon, Adam Porter, Arvind S. Krishna, Douglas C. Schmidt, Aniruddha Gokhale, and Balachandran Natarajan, "Preserving Distributed System's Critical Properties–A Model-Driven Approach," *IEEE Software*, Nov/Dec 2004.

Cemal Yilmaz, Arvind Krishna, Atif Memon, Adam Porter, Douglas C. Schmidt, Aniruddha Gokhale, and Bala Natarajan, "Main Effects Screening: A Distributed Continuous Quality Assurance Process for Monitoring Performance Degradation in Evolving Software Systems," Proceedings of the 27th International Conference on Software Engineering, St. Louis, MO, May 15-21, 2005.

Atif Memon, Adam Porter, Cemal Yilmaz, Adithya Nagarajan, Douglas C. Schmidt, and Bala Natarajan, "Skoll: Distributed Continuous Quality Assurance," Proceedings of the 26th IEEE/ACM International Conference on Software Engineering (ICSE), Edinburgh, Scotland, May 2004.

4) Mobile Cloud Computing Apps

The confluence of multi-core and distributed-core processors, inexpensive mass storage, ubiquitous wireless connectivity, and commodity software platforms is driving the need for software engineers and programmers who understand how to develop concurrent and networked software for mobile devices that connect to cloud computing platforms. Despite many improvements in processors, storage, and networks, however, developing quality software on-time and on-budget remains hard. Moreover, developing high quality reusable concurrent and networked software apps and services is even harder. Our work in this area focuses on how to apply patterns, pattern languages, and frameworks to alleviate the complexity of developing concurrent and networked software for mobile devices via the use of object-oriented design techniques, Java programming language features, and Android middleware. Representative publications related to mobile cloud computing as part of our R&D activities include:

1. Violetta Vylegzhanina, Douglas C. Schmidt, Pamela Hull, Janice S. Emerson, Meghan E. Quirk, and Shelagh Mulvaney, "Helping Children Eat Well Via Mobile Software Technologies," Proceedings of the Second International Workshop on Mobile Development Lifecycle, October 21st, 2015, Portland, OR.

2. Jules White, Douglas C. Schmidt, and Mani Golparvar-Fard, "Applications of Augmented Reality," IEEE Proceedings Special issue on Applications of Augmented Reality, Vol 102, No. 2., February 2014, pp. 120-123.

3. Chris Thompson, Jules White, and Douglas C. Schmidt, "Analyzing Mobile Application Software Power Consumption via Model-Driven Engineering," Advances and Applications in Model-Driven Software Engineering, edited by Vicente Garcia Diaz, IGI Global, 2013.

4. Zach McCormick and Douglas C. Schmidt, "Data Synchronization Patterns in Mobile Application Design," Proceedings of the Pattern Languages of Programs (PLoP) 2012 conference, October 19-21, Tucson, Arizona.

5. Benjamin Gotow, Krzysztof Zienkiewicz, Jules White, and Douglas C. Schmidt, "Addressing Challenges in Delivering Augmented Reality Applications to Smartphones," Proceedings of the Third International ICST Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications (Mobilware 2010), June 30-July 2, 2010, Chicago, IL.

6. Chris Thompson, Jules White, Brian Dougherty, Adam Albright, and Douglas C. Schmidt, "Using Smartphones and Wireless Mobile Networks to Detect Car Accidents and Provide Situational Awareness to Emergency Responders," Proceedings of the Third International ICST Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications (Mobilware 2010), June 30-July 2, 2010, Chicago, IL.

7. Jules White, Chris Thompson, Hamilton Turner, Brian Dougherty, and Douglas C. Schmidt, WreckWatch: Automatic Traffic Accident Detection and Notification with Smartphones, Journal of Mobile Networks and Applications, Volume 16 Issue 3, July 2011, Pages 285-303.

8. Jules White, Christin Groba, Sibohan Clarke, Brian Dougherty, Chris Thompson, and Douglas C. Schmidt, "R&D Challenges and Solutions for Mobile Cyber-Physical Applications and Supporting Internet Services," the Springer Journal of Internet Services and Applications, Volume 1, Number 1, 2010, pp. 45-56.

9. Hamilton Turner, Jules White, Brian Dougherty, and Douglas C. Schmidt, "Building Mobile Sensor Networks Using Smartphones and Web Services: Ramifications and Development Challenges," Handbook of Research on Mobility and Computing: Evolving Technologies and Ubiquitous Impacts, edited by Maria Manuela Cruz-Cunha and Fernando Moreira, IGI Global, Hershey, PA, USA 2009.

III. APPENDIX B: MATERIALS CONSIDERED

- <USES-SDK>, ANDROID, http://developer.android.com/guide/topics/manifest/uses-sdk-element.html (last visited January 8, 2016).
- ABSTRACT WINDOW TOOLKIT, ORACLE, http://docs.oracle.com/javase/7/docs/technotes/guides/awt/index.html (last visited January 7, 2016).
- ANROID, http://developer.android.com/images/system-architecture.jpg (last vsiited January 8, 2016).
- ANDROID COMPATIBILITY, ANDROID, https://source.android.com/compatibility/ (last visited January 7, 2016).
- ANDROID SDK, TECHOPEDIA, https://www.techopedia.com/definition/4220/android-sdk (last visited January 7, 2016).
- ANNOTATIONS, ORACLE, http://docs.oracle.com/javase/7/docs/technotes/guides/language/annotations.html (last visited January 8, 2016).
- ART AND DALVIK, ANDROID, https://source.android.com/devices/tech/dalvik (last visited January 7, 2016).
- CLASS CHARSET, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/nio/charset/Charset.html (last visited January 8, 2016).
- CLASS FONT, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/awt/Font.html (last visited January 8, 2016).
- COMPILER DEFINITION BY THE LINUX INFORMATION PROJECT (LINFO), http://www.linfo.org/compiler.html (last visited January 7, 2016).
- DALVIK EXECUTABLE FORMAT, ANDROID, https://source.android.com/devices/tech/dalvik/dex-format.html (last visited January 7, 2016).
- Dan Morrill, *Announcing the Android 1.0 SDK, release 1*, ANDROID DEVELOPERS BLOG (Sep. 23, 2008), http://android-developers.blogspot.com/2008/09/announcing-android-10-sdk-release-1.html.
- Declaration of Mark Reinhold, including Exhibits (Jan 7th 2016).
- ENUM TYPES (THE JAVA™ TUTORIALS > LEARNING THE JAVA LANGUAGE > CLASSES AND OBJECTS), http://docs.oracle.com/javase/tutorial/java/javaOO/enum.html (last visited January 7, 2016).
- Expert Report of Robert Zeidman, including Exhibits (Jan 8th 2016).
- FIELDS, THE JAVA™ TUTORIALS, ORACLE, https://docs.oracle.com/javase/tutorial/reflect/member/field.html.
- J2EE VS. MICROSOFT .NET, IBM, http://www-01.ibm.com/software/smb/na/J2EE_vs_NET_History_and_Comparison.pdf (last visited January 7, 2016).
- James Gosling, Bill Joy, Guy Steele, Gilad Bracha & Alex Buckley, *The Java Language Specification,* Oracle (Feb 13, 2015), https://docs.oracle.com/javase/specs/jls/se8/jls8.pdf.
- James Gosling, *Java: An Overview,* DARTMOUTH (Feb. 1995), pp. 2-5 *available at:* http://www.cs.dartmouth.edu/~mckeeman/cs118/references/OriginalJavaWhitepaper.pdf.
- JAVA API SPECIFICATIONS, ORACLE, http://www.oracle.com/technetwork/java/api-141528.html (last visited January 7, 2016).
- JAVA BEANS FAQ: GENERAL QUESTIONS, ORACLE, http://www.oracle.com/technetwork/java/javase/faq-135947.html (last visited January 7, 2016).
- JAVA LANGUAGE AND VIRTUAL MACHINE SPECIFICATIONS, ORACLE,

https://docs.oracle.com/javase/specs/ (last visited January 7, 2016).

- Java ME, CDC 1.0.2 is defined in JSR-36. *See* http://download.oracle.com/otn-pub/jcp/7825-j2me_cdc-1.0a-mr-spec-oth-JSpec/j2me_cdc-1_0a-mr-spec.zip.
- Java ME, CDC 1.1.2 is defined in JSR-218. *See* http://download.oracle.com/otn-pub/jcp/cdc-1.1-fr-eval-oth-spec-JSpec/cdc-1_1-fr-spec.zip; *see also* http://docs.oracle.com/javame/config/cdc/ref-impl/cdc1.1.2/jsr218/
- Java ME, CLDC 1.0 is defined in JSR-30. *See* http://download.oracle.com/otn-pub/jcp/7587-j2me_cldc-1.0a-fr-spec-oth-JSpec/cldc-1_0a-spec.zip; *see also* https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.0/cldcapi.pdf.
- Java ME, CLDC 1.1 is defined in JSR-139. *See* http://download.oracle.com/otn-pub/jcp/7247-j2me_cldc-1.1-fr-spec-oth-JSpec/cldc-1_1-fr-spec.zip; *see also* http://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.1/jsr139/
- JAVA PROGRAMMING ENVIRONMENT AND THE JAVA RUNTIME ENVIRONMENT (JRE), ORACLE, https://docs.oracle.com/cd/E19455-01/806-3461/6jck06gqd/index.html (Last visited Jan. 05, 2016).
- JAVA REFLECTION API, ORACLE, http://docs.oracle.com/javase/7/docs/technotes/guides/reflection/index.html (last visited January 8, 2016).
- JAVA VIRTUAL MACHINE, BUSINESS DICTIONARY, http://www.businessdictionary.com/definition/Java-virtual-machine-JVM.html (last visited Jan 5, 2016).
- JAVABEANS FAQ: GENERAL QUESTIONS, ORACLE, http://www.oracle.com/technetwork/java/javase/faq-135947.html (last visited January 8, 2016).
- JAVASOFT SHIPS JAVA 1.0, TECH-INSIDER, http://tech-insider.org/java/research/1996/0123.html (last visited January 7, 2016).
- JAVA™ PLATFORM STANDARD EDITION 7 NAMES AND VERSIONS, ORACLE, http://www.oracle.com/technetwork/java/javase/jdk7-naming-418744.html (last visited Jan. 5, 2016).
- JDK 5.0 DOCUMENTATION, SUN, https://web.archive.org/web/20100330080522/http://java.sun.com/j2se/1.5.0/docs/index.html (last visited January 7, 2016).
- Jeff Friesen, Overcoming Android's Problems with JDK 7, Part 1, INFORMIT (Oct 29, 2012), http://www.informit.com/articles/article.aspx?p=1966023.
- Jeff Friesen, Overcoming Android's Problems with JDK 7, Part 2, INFORMIT (Nov 5, 2012), http://www.informit.com/articles/article.aspx?p=1966024.
- Laurence Vanhelsuwé, *Need a good set of abstract data structures? ObjectSpace's JGL packs a punch!,* JAVAWORLD (Jan. 1, 1997), http://www.javaworld.com/article/2076958/java-app-dev/need-a-good-set-of-abstract-data-structures--objectspace-s-jgl-packs-a-punch-.html.
- ART AND DALVIK | ANDROID OPEN SOURCE PROJECT, http://source.android.com/devices/tech/dalvik/index.html (last visited January 7, 2016).
- Community Resources TCK Tools and Documentation, Java Community Process, https://jcp.org/en/resources/tdk, (last visited January 7, 2016), for a page on the Java Community Process (JCP)
- Java ACE, Vanderbilt, http://www.dre.vanderbilt.edu/JACE/ (for access to Java ACE downloads).
- JAVA COMPATIBILITYKIT 6B USER'S GUIDE, OPENJDK, http://openjdk.java.net/groups/conformance/docs/JCK6bUsersGuide/JCK6b_Users_Guide.pdf, (last visited January 7, 2016). for a copy of the user's guide for version 6b of the Java Compatibility Kit.

- LESSON: ANNOTATIONS (THE JAVA™ TUTORIALS > LEARNING THE JAVA LANGUAGE), https://docs.oracle.com/javase/tutorial/java/annotations/ (last visited January 7, 2016).
- MACHINE CODE DEFINITION, THE LINUX INFORMATION PROJECT, http://www.linfo.org/machine_code.html.
- OBTAINING ACE, TAO, CIAO, AND DANCE, VANDERBILT, http://download.dre.vanderbilt.edu/ (for access to ACE, TAO and CIAO downloads). *See* JAVA ACE, VANDERBILT, http://www.dre.vanderbilt.edu/JACE/ (for access to Java ACE downloads).
- ORACLE AND THE COMMUNITY CELEBRATE 20 YEARS OF JAVA, MARKET WIRED, http://www.marketwired.com/press-release/oracle-and-the-community-celebrate-20-years-of-java-nyse-orcl-2021578.htm (last visited January 8, 2016).
- Oracle's JRockit virtual machine product support specifically emphasizes compatibility across different hardware systems as well as between versions. *See* ORACLE JROCKIT PRODUCT SUPPOORT INFORMATION, ORACLE, http://docs.oracle.com/cd/E13150_01/jrockit_jvm/jrockit/jrdocs/suppPlat/prodsupp.html, (last visited Jan. 7, 2016).
- PACKAGE JAVA.IO DESCRIPTION, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/io/package-summary.html#package_description (last accessed January 8, 2016).
- PACKAGE JAVA.LANG, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/lang/package-summary.html (last visited January 8, 2016).
- PACKAGE JAVA.LANG.REF, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/lang/ref/package-summary.html (last visited January 8, 2016).
- PACKAGE JAVA.LANG.REFLECT, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/lang/reflect/package-summary.html (last visited January 7, 2016).
- PACKAGE JAVA.NET, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/net/package-summary.html (last visited January 7, 2016).
- PACKAGE JAVA.NIO.CHANNELS, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/nio/channels/package-summary.html (last visited January 8, 2016).
- PACKAGE JAVA.NIO.CHARSET.SPI, ORACLE, http://docs.oracle.com/javase/7/docs/api/java/nio/charset/spi/package-summary.html (last visited January 8, 2016).
- PACKAGE JAVA.SECURITY, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/security/package-summary.html (last visited January 7, 2016).
- PACKAGE JAVA.SECURITY.CERT DESCRIPTION, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/security/cert/package-summary.html#package_description (last visited January 7, 2016).
- PACKAGE JAVA.SQL, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/sql/package-summary.html (last visited January 7, 2016).
- PACKAGE JAVA.TEXT, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/text/package-summary.html (last visited January 8, 2016).
- PACKAGE JAVA.UTIL, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/util/package-summary.html (last visited January 8, 2016).
- PACKAGE JAVABEANS, ORACLE, http://docs.oracle.com/javase/7/docs/api/java/beans/package-summary.html (last visited January 8, 2016).
- PACKAGE JAVAX.CRYPTO, ORACLE, https://docs.oracle.com/javase/7/docs/api/javax/crypto/package-summary.html (last visited January 7, 2016).
- PACKAGE JAVAX.NET, ORACLE, http://docs.oracle.com/javase/7/docs/api/javax/net/package-summary.html (last visited January 7, 2016).

- PACKAGE JAVAX.SQL, ORACLE, http://docs.oracle.com/javase/7/docs/api/javax/sql/package-summary.html (last visited January 7, 2016).
- PAKCAGE JAVA.SECURITY.ACL, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/security/acl/Acl.html (last visited January 8, 2016).
- PORTABILITY, TECHOPEDIA, https://www.techopedia.com/definition/8921/portability.
- PROVIDING CONSTRUCTORS FOR YOURS CLASSES (THE JAVA™ TUTORIALS > LEARNING THE JAVA LANGUAGE > CLASSES AND OBJECTS) http://docs.oracle.com/javase/tutorial/java/javaOO/constructors.html (last visited January 7, 2016).
- Ron Amadeo, *Android 5.0 Lollipop, thoroughly reviewed,* ARS TECHNICA (Nov. 12, 1214), http://arstechnica.com/gadgets/2014/11/android-5-0-lollipop-thoroughly-reviewed.
- SOURCE CODE DEFINITION, THE LINUS INFORMATION PROJECT, http://www.linfo.org/source_code.html.
- SPRINGER/ANNALS OF TELECOMMUNICATIONS 2014: SPECIAL ISSUE ON MIDDLEWARE FOR INTERNET DISTRIBUTION IN THE CONTEXT OF CLOUD COMPUTING AND THE INTERNET OF THINGS, WIKICFP, http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=38110 (last visited January 7, 2016).
- SPRINGER/ANNALS OF TELECOMMUNICATIONS 2014: SPECIAL ISSUE ON MIDDLEWARE FOR INTERNET DISTRIBUTION IN THE CONTEXT OF CLOUD COMPUTING AND THE INTERNET OF THINGS, WIKICFP, http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=38110 (last visited January 7, 2016).
- Stephen Cass, *The 2015 Top Ten Programming Languages,* IEEE SPECTRUM (July 20, 2015) http://spectrum.ieee.org/computing/software/the-2015-top-ten-programming-languages.
- THE BEGINNER'S GUIDE TO ANDROID: ANDROID ARCHITECTURE, EDUREKA!,

  http://www.edureka.co/blog/beginners-guide-android-architecture (last visited January 8, 2016)

- THE HISTORY OF JAVA TECHNOLOGY, ORACLE, http://www.oracle.com/technetwork/java/javase/overview/javahistory-index-198355.html (last visited January 7, 2016).
- THE JAVA LANGUAGE ENVIRONMENT, http://www.oracle.com/technetwork/java/intro-141325.html#379 (last visited January 7, 2016).
- Tim Lindholm, Frank Yellin, Gilad Bracha & Alex Buckley, *The Java Virtual Machine Specification*, ORACLE (Feb. 13, 2015), https://docs.oracle.com/javase/specs/jvms/se8/jvms8.pdf
- TIM LINDHOLM & FRANK YELLIN, THE JAVA VIRTUAL MACHINE SPECIFICATION (Addison-Wesley Longman Publishing Co., Inc, 2d ed. 1999).
- United States Court of Appeals for the Federal Circuit, Oracle America Inc. v. Google Inc. 2013-1021, 1022, at 11.
- USING PACKAGE MEMBERS, ORACLE, https://docs.oracle.com/javase/tutorial/java/package/usepkgs.html (Last accessed on 01/07/2016).
- WHAT IS A SYSTEM IMAGE?, HOW-TO GEEK, http://www.howtogeek.com/192115/what-you-need-to-know-about-creating-system-image-backups/ (last visited Jan. 7, 2016).
- WHAT IS AN EXCEPTION?, THE JAVA™ TUTORIALS, ORACLE, https://docs.oracle.com/javase/tutorial/essential/exceptions/definition.html (last visited January 8, 2016).
- WRITING FINAL CLASSES AND METHODS (THE JAVA™ TUTORIALS > LEARNING THE JAVA LANGUAGE > INTERFACES AND INHERITANCE) http://docs.oracle.com/javase/tutorial/java/IandI/final.html (last visited January 7, 2016).

IV. APPENDIX C: THE JAVA CLASS LIBRARIES

Trial Exhibit 1028, Page 1 of 1

V. APPENDIX D: EVOLUTION OF JAVA APIS

Java API Packages JDK Versions 1-8

Package	1.0	1.1	1.2	1.3	1.4	1.5	6.0	7.0	8.0
sun.tools.debug	X								
java.awt.peer	X								
java.applet	X	X	X	X	X	X	X	X	X
java.awt.image	X	X	X	X	X	X	X	X	X
java.awt	X	X	X	X	X	X	X	X	X
java.net	X	X	X	X	X	X	X	X	X
java.util	X	X	X	X	X	X	X	X	X
java.io	X	X	X	X	X	X	X	X	X
java.lang	X	X	X	X	X	X	X	X	X
java.awt.datatransfer		X	X	X	X	X	X	X	X
java.awt.event		X	X	X	X	X	X	X	X
java.beans		X	X	X	X	X	X	X	X
java.lang.reflect		X	X	X	X	X	X	X	X
java.math		X	X	X	X	X	X	X	X
java.rmi		X	X	X	X	X	X	X	X
java.rmi.dgc		X	X	X	X	X	X	X	X
java.rmi.registry		X	X	X	X	X	X	X	X
java.rmi.server		X	X	X	X	X	X	X	X
java.security		X	X	X	X	X	X	X	X
java.security.acl		X	X	X	X	X	X	X	X
java.security.interfaces		X	X	X	X	X	X	X	X
java.sql		X	X	X	X	X	X	X	X
java.text		X	X	X	X	X	X	X	X
java.util.zip		X	X	X	X	X	X	X	X
java.awt.color			X	X	X	X	X	X	X
java.awt.dnd			X	X	X	X	X	X	X
java.awt.font			X	X	X	X	X	X	X
java.awt.geom			X	X	X	X	X	X	X
java.awt.im			X	X	X	X	X	X	X
java.awt.image.renderable			X	X	X	X	X	X	X
java.awt.print			X	X	X	X	X	X	X
java.beans.beancontext			X	X	X	X	X	X	X
java.lang.ref			X	X	X	X	X	X	X
java.rmi.activation			X	X	X	X	X	X	X
java.security.cert			X	X	X	X	X	X	X
java.security.spec			X	X	X	X	X	X	X
java.util.jar			X	X	X	X	X	X	X

Package	1.0	1.1	1.2	1.3	1.4	1.5	6.0	7.0	8.0
javax.accessibility			X	X	X	X	X	X	X
javax.swing			X	X	X	X	X	X	X
javax.swing.border			X	X	X	X	X	X	X
javax.swing.colorchooser			X	X	X	X	X	X	X
javax.swing.event			X	X	X	X	X	X	X
javax.swing.filechooser			X	X	X	X	X	X	X
javax.swing.plaf			X	X	X	X	X	X	X
javax.swing.plaf.basic			X	X	X	X	X	X	X
javax.swing.plaf.metal			X	X	X	X	X	X	X
javax.swing.plaf.multi			X	X	X	X	X	X	X
javax.swing.table			X	X	X	X	X	X	X
javax.swing.text			X	X	X	X	X	X	X
javax.swing.text.html			X	X	X	X	X	X	X
javax.swing.text.html.parser			X	X	X	X	X	X	X
javax.swing.text.rtf			X	X	X	X	X	X	X
javax.swing.tree			X	X	X	X	X	X	X
javax.swing.undo			X	X	X	X	X	X	X
org.omg.CORBA			X	X	X	X	X	X	X
org.omg.CORBA.DynAnyPackage			X	X	X	X	X	X	X
org.omg.CORBA.ORBPackage			X	X	X	X	X	X	X
org.omg.CORBA.portable			X	X	X	X	X	X	X
org.omg.CORBA.TypeCodePackage			X	X	X	X	X	X	X
org.omg.CosNaming			X	X	X	X	X	X	X
org.omg.CosNaming.NamingContextPackage			X	X	X	X	X	X	X
java.awt.im.spi				X	X	X	X	X	X
javax.naming				X	X	X	X	X	X
javax.naming.directory				X	X	X	X	X	X
javax.naming.event				X	X	X	X	X	X
javax.naming.ldap				X	X	X	X	X	X
javax.naming.spi				X	X	X	X	X	X
javax.rmi				X	X	X	X	X	X
javax.rmi.CORBA				X	X	X	X	X	X
javax.sound.midi				X	X	X	X	X	X
javax.sound.midi.spi				X	X	X	X	X	X
javax.sound.sampled				X	X	X	X	X	X
javax.sound.sampled.spi				X	X	X	X	X	X
javax.transaction				X	X	X	X	X	X
org.omg.CORBA_2_3				X	X	X	X	X	X
org.omg.CORBA_2_3.portable				X	X	X	X	X	X
org.omg.SendingContext				X	X	X	X	X	X
org.omg.stub.java.rmi				X	X	X	X	X	X

Package	1.0	1.1	1.2	1.3	1.4	1.5	6.0	7.0	8.0
java.nio					X	X	X	X	X
java.nio.channels					X	X	X	X	X
java.nio.channels.spi					X	X	X	X	X
java.nio.charset					X	X	X	X	X
java.nio.charset.spi					X	X	X	X	X
java.util.logging					X	X	X	X	X
java.util.prefs					X	X	X	X	X
java.util.regex					X	X	X	X	X
javax.crypto					X	X	X	X	X
javax.crypto.interfaces					X	X	X	X	X
javax.crypto.spec					X	X	X	X	X
javax.imageio					X	X	X	X	X
javax.imageio.event					X	X	X	X	X
javax.imageio.metadata					X	X	X	X	X
javax.imageio.plugins.jpeg					X	X	X	X	X
javax.imageio.spi					X	X	X	X	X
javax.imageio.stream					X	X	X	X	X
javax.net					X	X	X	X	X
javax.net.ssl					X	X	X	X	X
javax.print					X	X	X	X	X
javax.print.attribute					X	X	X	X	X
javax.print.attribute.standard					X	X	X	X	X
javax.print.event					X	X	X	X	X
javax.security.auth					X	X	X	X	X
javax.security.auth.callback					X	X	X	X	X
javax.security.auth.kerberos					X	X	X	X	X
javax.security.auth.login					X	X	X	X	X
javax.security.auth.spi					X	X	X	X	X
javax.security.auth.x500					X	X	X	X	X
javax.security.cert					X	X	X	X	X
javax.sql					X	X	X	X	X
javax.transaction.xa					X	X	X	X	X
javax.xml.parsers					X	X	X	X	X
javax.xml.transform					X	X	X	X	X
javax.xml.transform.dom					X	X	X	X	X
javax.xml.transform.sax					X	X	X	X	X
javax.xml.transform.stream					X	X	X	X	X
org.ietf.jgss					X	X	X	X	X
org.omg.CosNaming.NamingContextExtPackage					X	X	X	X	X
org.omg.Dynamic					X	X	X	X	X
org.omg.DynamicAny					X	X	X	X	X

Package	1.0	1.1	1.2	1.3	1.4	1.5	6.0	7.0	8.0
org.omg.DynamicAny.DynAnyFactoryPackage					X	X	X	X	X
org.omg.DynamicAny.DynAnyPackage					X	X	X	X	X
org.omg.IOP					X	X	X	X	X
org.omg.IOP.CodecFactoryPackage					X	X	X	X	X
org.omg.IOP.CodecPackage					X	X	X	X	X
org.omg.Messaging					X	X	X	X	X
org.omg.PortableInterceptor					X	X	X	X	X
org.omg.PortableInterceptor.ORBInitInfoPackage					X	X	X	X	X
org.omg.PortableServer					X	X	X	X	X
org.omg.PortableServer.CurrentPackage					X	X	X	X	X
org.omg.PortableServer.POAManagerPackage					X	X	X	X	X
org.omg.PortableServer.POAPackage					X	X	X	X	X
org.omg.PortableServer.portable					X	X	X	X	X
org.omg.PortableServer.ServantLocatorPackage					X	X	X	X	X
org.w3c.dom					X	X	X	X	X
org.xml.sax					X	X	X	X	X
org.xml.sax.ext					X	X	X	X	X
org.xml.sax.helpers					X	X	X	X	X
java.lang.annotation						X	X	X	X
java.lang.instrument						X	X	X	X
java.lang.management						X	X	X	X
java.util.concurrent						X	X	X	X
java.util.concurrent.atomic						X	X	X	X
java.util.concurrent.locks						X	X	X	X
javax.imageio.plugins.bmp						X	X	X	X
javax.management						X	X	X	X
javax.management.loading						X	X	X	X
javax.management.modelmbean						X	X	X	X
javax.management.monitor						X	X	X	X
javax.management.openmbean						X	X	X	X
javax.management.relation						X	X	X	X
javax.management.remote						X	X	X	X
javax.management.remote.rmi						X	X	X	X
javax.management.timer						X	X	X	X
javax.rmi.ssl						X	X	X	X
javax.security.sasl						X	X	X	X
javax.sql.rowset						X	X	X	X
javax.sql.rowset.serial						X	X	X	X
javax.sql.rowset.spi						X	X	X	X

Package	1.0	1.1	1.2	1.3	1.4	1.5	6.0	7.0	8.0
javax.swing.plaf.synth						X	X	X	X
javax.xml						X	X	X	X
javax.xml.datatype						X	X	X	X
javax.xml.namespace						X	X	X	X
javax.xml.validation						X	X	X	X
javax.xml.xpath						X	X	X	X
org.w3c.dom.bootstrap						X	X	X	X
org.w3c.dom.events						X	X	X	X
org.w3c.dom.ls						X	X	X	X
javax.activity						X	X	X	X
java.text.spi							X	X	X
java.util.spi							X	X	X
javax.activation							X	X	X
javax.annotation							X	X	X
javax.annotation.processing							X	X	X
javax.jws							X	X	X
javax.jws.soap							X	X	X
javax.lang.model							X	X	X
javax.lang.model.element							X	X	X
javax.lang.model.type							X	X	X
javax.lang.model.util							X	X	X
javax.script							X	X	X
javax.tools							X	X	X
javax.xml.bind							X	X	X
javax.xml.bind.annotation							X	X	X
javax.xml.bind.annotation.adapters							X	X	X
javax.xml.bind.attachment							X	X	X
javax.xml.bind.helpers							X	X	X
javax.xml.bind.util							X	X	X
javax.xml.crypto							X	X	X
javax.xml.crypto.dom							X	X	X
javax.xml.crypto.dsig							X	X	X
javax.xml.crypto.dsig.dom							X	X	X
javax.xml.crypto.dsig.keyinfo							X	X	X
javax.xml.crypto.dsig.spec							X	X	X
javax.xml.soap							X	X	X
javax.xml.stream							X	X	X
javax.xml.stream.events							X	X	X
javax.xml.stream.util							X	X	X
javax.xml.transform.stax							X	X	X
javax.xml.ws							X	X	X
javax.xml.ws.handler							X	X	X

Package	1.0	1.1	1.2	1.3	1.4	1.5	6.0	7.0	8.0
javax.xml.ws.handler.soap							X	X	X
javax.xml.ws.http							X	X	X
javax.xml.ws.soap							X	X	X
javax.xml.ws.spi							X	X	X
javax.xml.ws.wsaddressing							X	X	X
java.lang.invoke								X	X
java.nio.file								X	X
java.nio.file.attribute								X	X
java.nio.file.spi								X	X
javax.swing.plaf.nimbus								X	X
javax.xml.ws.spi.http								X	X
java.time									X
java.time.chrono									X
java.time.format									X
java.time.temporal									X
java.time.zone									X
java.util.function									X
java.util.stream									X
org.w3c.dom.views									X
Total	9	22	59	76	135	166	203	209	217

VI. APPENDIX E: GLOSSARY OF TERMS

Term	Definition
Abstract Methods and Classes	An abstract method is a method declared without an implementation. An abstract class is a class with one or more abstract methods. An abstract class cannot be directly instantiated and must be extended by another subclass that contains implementations for the abstract methods.
Android Software Development Kit ("SDK")	A set of development tools that are used to develop apps on the Android platform, including required libraries, debugger, emulator, relevant documentation for Android APIs, sample source code, and tutorials for the Android OS.[1]
Annotation	Java annotations are a form of metadata that provides information about a program that is not part of the program itself. Annotations have no direct effect on the operation of the code they annotate. Annotations can provide information used by the compiler to detect errors or suppress warnings; provide information to software tools used to generate code or other files; or provide information available for examination at runtime.[2]
Architecture-neutral	The property of being easily transported to a diversity of hardware platforms and software operating environments.[3]
Class	A class is a blueprint from which individual objects (instances) are created. Classes contain members that include methods to describe an object's behavior and fields that store values to capture an object's state.[4]
Compiler	A compiler is a program that converts source code written in some high level programming language into another target computer language. The target computer language is usually in binary form, such as bytecode or machine code that can be executed by a virtual machine or physical machine (e.g., by a computer).[5]
Constructor	A Java constructor is a member of a class that looks like a method except that it uses the name of the class. A constructor is invoked to create objects using the class blueprint.[6]
Dalvik/ART	ART (Android runtime) and Dalvik are "managed runtimes," or virtual machines, that can execute Dex bytecode on Android.[7]
Enum	An enum is a special data type that enables a variable to be a set of predefined constants. The variable must be equal to one of the values in the set. A common example is compass directions, with values NORTH,

[1] ANDROID SDK, TECHOPEDIA, https://www.techopedia.com/definition/4220/android-sdk (last visited January 7, 2016).
[2] LESSON: ANNOTATIONS (THE JAVA™ TUTORIALS > LEARNING THE JAVA LANGUAGE), https://docs.oracle.com/javase/tutorial/java/annotations/ (last visited January 7, 2016).
[3] THE JAVA LANGUAGE ENVIRONMENT, http://www.oracle.com/technetwork/java/intro-141325.html#379 (last visited January 7, 2016).
[4] See, e.g., CLASSES AND OBJECTS: CHAPTER 2 OF OBJECTS AND JAVA, http://www.artima.com/objectsandjava/webuscript/ClassesObjects1.html (last visited January 7, 2016).
[5] COMPILER DEFINITION BY THE LINUX INFORMATION PROJECT (LINFO), http://www.linfo.org/compiler.html (last visited January 7, 2016).
[6] PROVIDING CONSTRUCTORS FOR YOURS CLASSES (THE JAVA™ TUTORIALS > LEARNING THE JAVA LANGUAGE > CLASSES AND OBJECTS) http://docs.oracle.com/javase/tutorial/java/javaOO/constructors.html (last visited January 7, 2016).
[7] ART AND DALVIK | ANDROID OPEN SOURCE PROJECT, http://source.android.com/devices/tech/dalvik/index.html (last visited January 7, 2016).

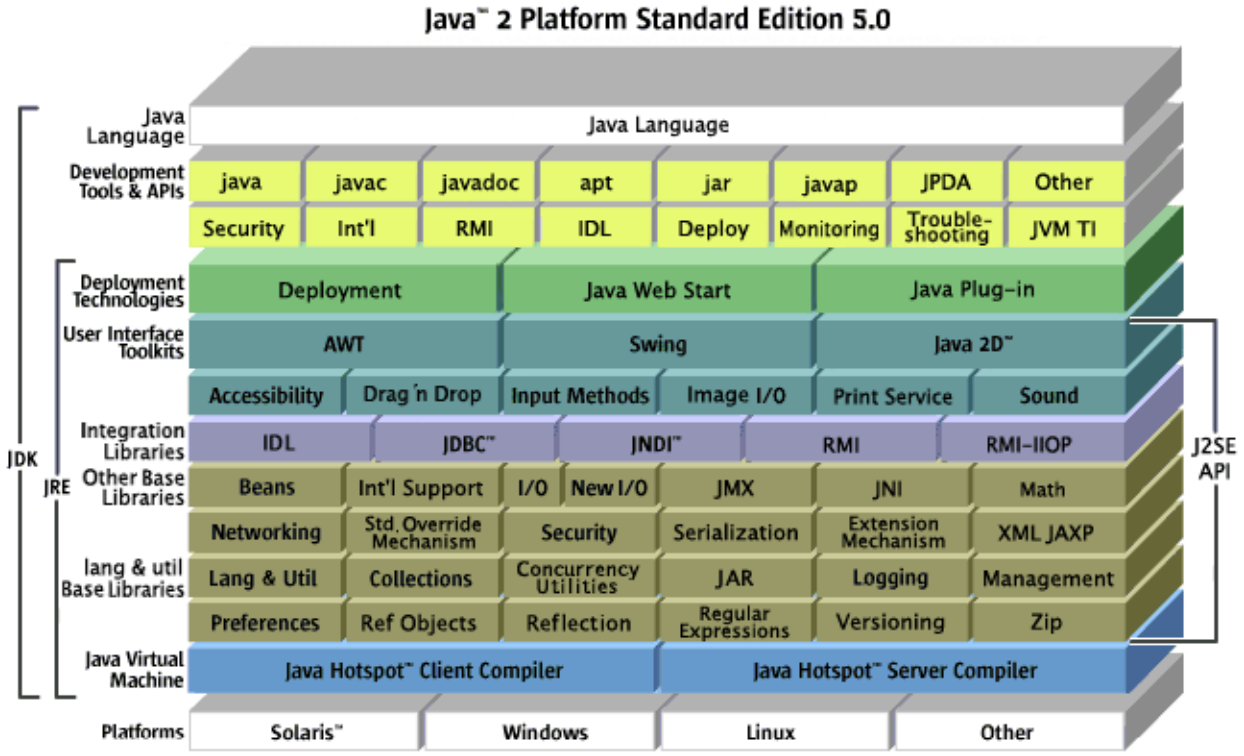
Term	Definition
	SOUTH, EAST, and WEST.[8]
Error	An error is typically a catastrophic event from which a program cannot recover, and usually results in the program terminating. For example, a computer running out of memory is an example of an error.
Exception	An exception is an event that occurs during the execution of a program that disrupts the normal flow of instructions.[9]
Field	Fields are member variables in a class.[10]
Final	Final is a keyword in Java used to declare aspects of an entity. A final class is a class that cannot be subclassed. A final method is a method that cannot be overridden in subclasses. A final field is a field that can be assigned only once.[11]
Implementing code	Implementing code is software code that defines a method specified by declaring code.
Import statements	Import statements are used by a package to specify the members of outside public packages and classes that are used by the package containing the import statement.[12]
Inheritance	Inheritance in object-oriented programming allows classes to inherit commonly used state (fields) and behavior (methods) from other classes (called superclasses).[13]
Interface	An interface in Java is a reference type similar to a class that can only contain constants, method signatures, default methods, static methods and nested types. Method signatures of interfaces do not include implementing code, and hence interfaces can not be used to create objects. Developers write classes that implement interfaces so that their software can interact.[14]
Java API packages	The Java API packages are a set of prewritten Java software packages, including classes and methods, that developers can use in lieu of writing their own software for similar behavior
Java bytecode	Java bytecode is the code created by the Java compiler from a developer's source code. Bytecode provides instructions to a Java Virtual Machine describing the behavior of the source code.
Java class file	A Java class file is a file produced by the Java compiler from the source code file of the associated Java class (i.e., a *.java file).
Java Development Kit ("JDK")	The JDK is a development environment for building applications that includes the Java Runtime Environment, Java Compiler, and the Java API packages.[15]
Java Platform	The Java platform includes resources such as the Java Language, the Java

[8] ENUM TYPES (THE JAVA™ TUTORIALS > LEARNING THE JAVA LANGUAGE > CLASSES AND OBJECTS), http://docs.oracle.com/javase/tutorial/java/javaOO/enum.html (last visited January 7, 2016).
[9] WHAT IS AN EXCEPTION?, THE JAVA™ TUTORIALS, ORACLE, https://docs.oracle.com/javase/tutorial/essential/exceptions/definition.html.
[10] FIELDS, THE JAVA™ TUTORIALS, ORACLE, https://docs.oracle.com/javase/tutorial/reflect/member/field.html.
[11] WRITING FINAL CLASSES AND METHODS (THE JAVA™ TUTORIALS > LEARNING THE JAVA LANGUAGE > INTERFACES AND INHERITANCE) http://docs.oracle.com/javase/tutorial/java/IandI/final.html (last visited January 7, 2016).
[12] USING PACKAGE MEMBERS, ORACLE, https://docs.oracle.com/javase/tutorial/java/package/usepkgs.html (Last accessed on 01/07/2016).
[13] Jeff Friesen, *Java 101: Inheritance in Java, Part 1*, JAVAWORLD (Oct. 22, 2015), http://www.javaworld.com/article/2987426/core-java/java-101-inheritance-in-java-part-1.html.
[14] Jacob Jenkov, *Java Interfaces*, tutorials.jenkov.com/java/interfaces.html.
[15] *See, e.g.*, JAVA™ PLATFORM STANDARD EDITION 7 NAMES AND VERSIONS, ORACLE, http://www.oracle.com/technetwork/java/javase/jdk7-naming-418744.html (last visited Jan. 5, 2016).

Term	Definition
	Runtime Environment, the Java Compiler, the Java Virtual Machine, and the Java API packages.
Java Runtime Environment (JRE)	The JRE provides the libraries, Java virtual machine, and other components necessary to run progams written in the Java language.[16]
Java Virtual Machine (JVM)	The JVM is an abstract computing machine that can be used to execute Java bytecode and can be ported to different operating platforms to provide hardware and operating system independence.[17]
Machine code	Machine code is a set of binary instructions that can be executed by a computing device and that is specific to the type of computing device.[18]
Method	In object-oriented programming, a method is a member of a class that defines behavior of the class.
Native	Native is a keyword in Java used to indicate that the implementation for a method is written in a language other than Java.
Packages	A package in Java is a namespace that organizes a set of related types, including classes and interfaces.
Portability	Portability in Java refers to the ability to execute Java bytecode on instances of a Java Virtual Machine across a diverse set of hardware and operating system environments.
Source code	Source code is computer software (instructions) written by software developers in a human-readable, high-level programming language, like Java.
Static	Static is a keyword in the Java programming language that is used to create fields and methods that belong to the class (meaning they are accessible by all instances of that class, i.e. objects created from the class), rather than belonging to a single instance of the class.
System image files	Android system image files are files containing executable software code required to run Android on a computing device.

[16] JAVA PROGRAMMING ENVIRONMENT AND THE JAVA RUNTIME ENVIRONMENT (JRE), ORACLE, https://docs.oracle.com/cd/E19455-01/806-3461/6jck06gqd/index.html (Last visited Jan. 05, 2016).
[17] JAVA VIRTUAL MACHINE, BUSINESS DICTIONARY, http://www.businessdictionary.com/definition/Java-virtual-machine-JVM.html (last visited Jan 5, 2016).
[18] MACHINE CODE DEFINITION, THE LINUX INFORMATION PROJECT, http://www.linfo.org/machine_code.html.

VII. APPENDIX F: JAVA SE 5.0 PLATFORM[19]



[19] JDK 5.0 DOCUMENTATION, ORACLE, http://docs.oracle.com/javase/1.5.0/docs/ (last visited January 8, 2016).

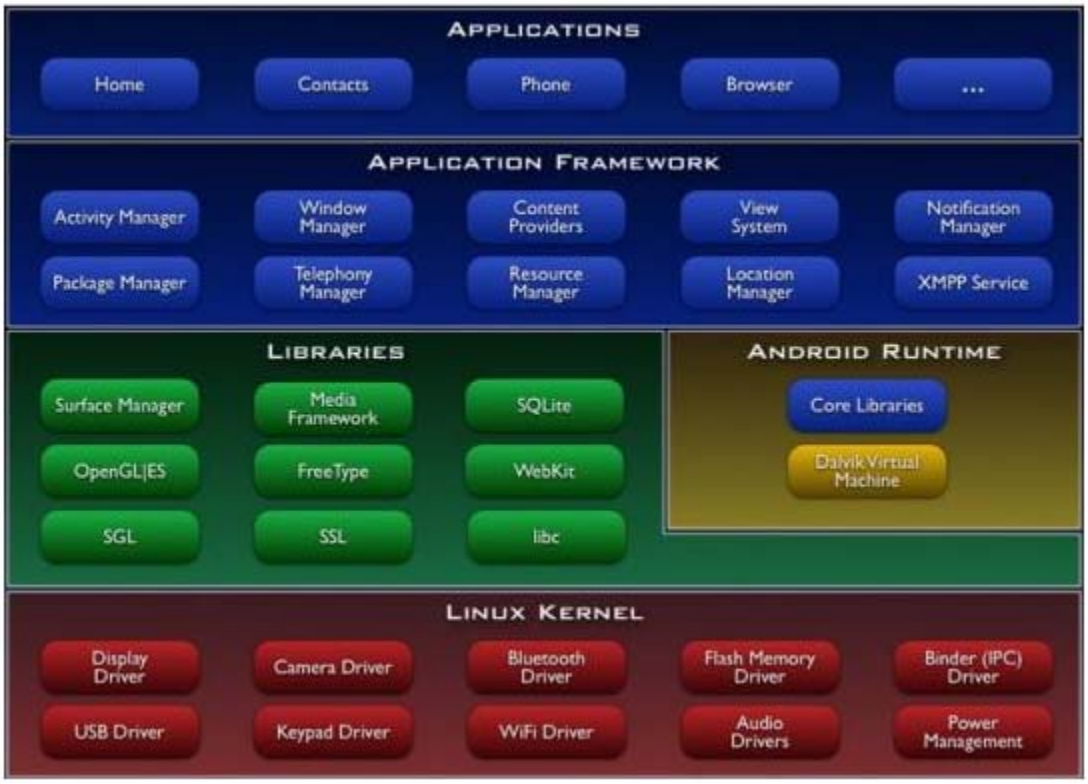
VIII. APPENDIX G: PACKAGES IN JAVA SE 5 AND COPIED API PACKAGES

Packages In Java SE 5 - 37 Copied Java Packages In Green

java.applet	java.awt.image.renderable	java.math	java.rmi.server	java.util.concurrent.locks	javax.imageio	javax.management.openmbean	javax.net.ssl	javax.security.auth.login	javax.sql.rowset.serial
java.awt	java.awt.print	java.net	java.security	java.util.jar	javax.imageio.event	javax.management.relation	javax.print	javax.security.auth.spi	javax.sql.rowset.spi
java.awt.color	java.beans	java.nio	java.security.acl	java.util.logging	javax.imageio.metadata	javax.management.remote	javax.print.attribute	javax.security.auth.x500	javax.swing
java.awt.datatransfer	java.beans.beancontext	java.nio.channels	java.security.cert	java.util.prefs	javax.imageio.plugins.bmp	javax.management.remote.rmi	javax.print.attribute.standard	javax.security.cert	javax.swing.border
java.awt.dnd	java.io	java.nio.channels.spi	java.security.interfaces	java.util.regex	javax.imageio.plugins.jpeg	javax.management.timer	javax.print.event	javax.security.sasl	javax.swing.colorchooser
java.awt.event	java.lang	java.nio.charset	java.security.spec	java.util.zip	javax.imageio.spi	javax.naming	javax.rmi	javax.sound.midi	javax.swing.event
java.awt.font	java.lang.annotation	java.nio.charset.spi	java.sql	javax.accessibility	javax.imageio.stream	javax.naming.directory	javax.rmi.CORBA	javax.sound.midi.spi	javax.swing.filechooser
java.awt.geom	java.lang.instrument	java.rmi	java.text	javax.activity	javax.management	javax.naming.event	javax.rmi.ssl	javax.sound.sampled	javax.swing.plaf
java.awt.im	java.lang.management	java.rmi.activation	java.util	javax.crypto	javax.management.loading	javax.naming.ldap	javax.security.auth	javax.sound.sampled.spi	javax.swing.plaf.basic
java.awt.im.spi	java.lang.ref	java.rmi.dgc	java.util.concurrent	javax.crypto.interfaces	javax.management.modelmbean	javax.naming.spi	javax.security.auth.callback	javax.sql	javax.swing.plaf.metal
java.awt.image	java.lang.reflect	java.rmi.registry	java.util.concurrent.atomic	javax.crypto.spec	javax.management.monitor	javax.net	javax.security.auth.kerberos	javax.sql.rowset	javax.swing.plaf.multi
org.omg.PortableServer.POAPackage	org.omg.PortableServer.CurrentPackage	org.omg.PortableServer.POAManagerPackage	org.omg.PortableServer.portable	org.omg.PortableServer.ServantLocatorPackage	org.omg.SendingContext	org.omg.stub.java.rmi	org.w3c.dom	org.w3c.dom.bootstrap	org.w3c.dom.events
org.omg.Dynamic	org.omg.DynamicAny	org.xml.sax.helpers	org.xml.sax.ext	org.omg.DynamicAny.DynAnyPackage	org.omg.IOP	org.omg.IOP.CodecFactoryPackage	org.omg.IOP.CodecPackage	org.omg.Messaging	org.omg.PortableInterceptor
org.omg.PortableServer	org.omg.CosNaming.NamingContextPackage	org.omg.CORBA_2_3	org.omg.CORBA_2_3.portable	org.omg.CORBA.DynAnyPackage	org.omg.CORBA.ORBPackage	org.omg.CORBA.portable	org.omg.CORBA.TypeCodePackage	org.omg.CosNaming	org.omg.DynamicAny.DynAnyFactoryPackage
javax.xml.namespace	javax.xml.parsers	javax.xml.transform	javax.xml.transform.dom	javax.xml.transform.sax	javax.xml.transform.stream	javax.xml.validation	javax.xml.xpath	org.ietf.jgss	org.omg.CORBA

javax.swing.tree	javax.swing.undo	javax.transaction	javax.transaction.xa	javax.xml	org.w3c.dom.ls	org.omg.PortableInterceptor.ORBInitInfoPackage	org.omg.CosNaming.NamingContextExtPackage	org.xml.sax	javax.xml.datatype
				javax.swing.text	javax.swing.table	javax.swing.plaf.synth	javax.swing.text.html	javax.swing.text.html.parser	javax.swing.text.rtf

IX. APPENDIX H: ANDROID PLATFORM[20]



[20] ANROID, http://developer.android.com/images/system-architecture.jpg (last vsiited January 8, 2016).

X. APPENDIX I: LIST OF COPIED ORACLE APIS

Copied Oracle API	API Purpose	First Android Version to Copy
java.awt.font	AWT (Abstract Window Toolkit) provides a set of native user interface components (such as the display of graphics and fonts), and an event-handling model (e.g. to handle user clicks in the user interface). It includes graphics and imaging tools to display shapes, and color, and to "render" fonts, i.e. to display text and other typographic features in a visible way (e.g. to display text on the screen with line breaks at appropriate places in the group of characters that make up the text). It also includes tools to facilitate flexible window layouts in a flexible way that does not depend on the window size or the screens resolution. It also enables "cut and paste" capabilities for the clipboard.[21] Java.awt.font more speficially supports fonts such as "TrueType" or "PostScript" and facilitates displaying text using the different typographic designs of various fonts, as well as using text attributes like "bold."[22]	1.0 (Level 1 - "Base")
java.beans	Provides code that aids developers in creating reusable software components for applications that use the JavaBeans™ architecture. JavaBeans™ is a portable, platform-independent component model that enables developers to write reusable components once and run them anywhere, and it can act as a bridge to other proprietary component models).[23] JavaBeans are typically deployed in a network or distributed application environment, and they provide ways for components to work together that do not necessarily "know about each other" in advance. This package also provides capabilities such as a *long term* "persistence" approach, i.e. a way to store the data (or state) associated with a component (bean) in a database or other data repository (e.g. using XML format).[24]	1.5 (Level 3 - "Cupcake")

[21] Abstract Window Toolkit (AWT), ORACLE, http://docs.oracle.com/javase/7/docs/technotes/guides/awt/index.html (last visited January 8, 2016).
[22] CLASS FONT, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/awt/Font.html (last visited January 8, 2016).
[23] JAVABEANS FAQ: GENERAL QUESTIONS, ORACLE, http://www.oracle.com/technetwork/java/javase/faq-135947.html (last visited January 8, 2016).

[24] PACKAGE JAVABEANS, ORACLE, http://docs.oracle.com/javase/7/docs/api/java/beans/package-summary.html (last visited January 8, 2016).

Copied Oracle API	API Purpose	First Android Version to Copy
java.io	Provides components for reading and writing data (input and output), such as reading or writing data from or to a file or other source located over a network. The classes of this package can take data representing various types, like characters, numbers or other information; turn them into a sequence of binary bits; stream them; and then reconstruct the appropriate type(s) of information (i.e. characters, numbers, etc…).[25]	1.0 (Level 1 - "Base")
java.lang	The package provides "system operations" that manage the dynamic loading of classes, the creation of new operating system processes or processor threads (to facilitate concurrent processing), the querying of the host environment for information such as the time of day, and the enforcement of security policies. The package also defines wrapper classes and data type conversions for certain data types that may be represented as objects, such as Boolean, Character, Integer, Long, Short, Float and Double, and defines objects, classes and defines state and behaviors such the ability of an object to compare itself to another object, to convert itself to a string, to wait (suspend execution) on the condition of a variable or notification, to notify other objects of the change in a condition variable, and to return the objects class name. In addition classes in this package enable capabilities for other classes such as cloning of an object, and garbage collection (i.e. freeing up the memory associated with an object when there are no longer any references to it).[26]	1.0 (Level 1 - "Base")
java.lang.annotation	Provides a facility for developers to add "meta data" to their code which serves the purpose of providing compiler instructions, build-time instructions, and runtime instructions. In other words, annotations affect the way tools and libraries treat a program; normally the annotations to not affect program execution. As an example, the "@deprecated" annotation can give a compiler instructions that the marked class, method or field has been deprecated and should no longer be used. The compiler will then issue a warning during	1.0 (Level 1 - "Base")

[25] PACKAGE JAVA.IO DESCRIPTION, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/io/package-summary.html#package_description (last accessed January 8, 2016).

[26] PACKAGE JAVA.LANG, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/lang/package-summary.html (last visited January 8, 2016).

Copied Oracle API	API Purpose	First Android Version to Copy
	compilation.[27]	
java.lang.ref	Provides component that enables interaction with the garbage collector. The package is used to create a wrapper around a "referent", i.e. an object that the reference points to, that allows the program to remain aware of the garbage collection status of an object without affecting the garbage collection process.[28]	1.0 (Level 1 - "Base")
java.lang.reflect	Provides component to obtain information programmatically about fields, methods and constructors of loaded classes, as well as to use this information to operate on underlying fields, methods and constructors. Classes in this package are often used by applications such as debuggers, interpreters, object inspectors, class browsers, and other services that need access to either the public members of a target object (based on its runtime class) or the members declared by a given class. For example, the classes in this package can be used to determine the length of an array represented by an array object.[29], [30]	1.0 (Level 1 - "Base")
java.net	Provides components for implementing networking applications, roughly divided into two sections: i) ability to deal with addresses like IP addresses, sockets (bi-directional communication mechanisms), and network interfaces; and ii) capabilities to deal with Universal Resource Identifiers, URLs, which represent Universal Resource Locators, and "connections" to the resource pointed to by URLs.[31] As one example, these classes include mechanisms for requesting and performing password authentication over a network.	1.0 (Level 1 - "Base")

[27] ANNOTATIONS, ORACLE, http://docs.oracle.com/javase/7/docs/technotes/guides/language/annotations.html (last visited January 8, 2016).

[28] PACKAGE JAVA.LANG.REF, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/lang/ref/package-summary.html (last visited January 8, 2016).

[29] PACKAGE JAVA.LANG.REFLECT, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/lang/reflect/package-summary.html (last visited January 8, 2016).

[30] JAVA REFLECTION API, ORACLE, http://docs.oracle.com/javase/7/docs/technotes/guides/reflection/index.html (last visited January 8, 2016).

[31] PACKAGE JAVA.NET, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/net/package-summary.html (last visited January 8, 2016).

Copied Oracle API	API Purpose	First Android Version to Copy
java.nio	Provides components for system input and output through data streams, serialization and the file system. Its capabilities are used for data intensive Input/Output operations as an extension to the capabilities provided by the java.io package.[32]	1.0 (Level 1 - "Base")
java.nio.channels	Provides components to define channels, which represent connections to entities that are capable of performing I/O operations, such as files and sockets, and defines selectors, for multiplexed, non-blocking I/O operations. Non-blocking means threads of computation do not have to wait for I/O operations to complete before continuing, while multiplexing means multiple connections can be used simultaneously using a "selector."[33]	1.0 (Level 1 - "Base")
java.nio.channels.spi	Provides service-provider components for the java.nio.channels package that is intended to be used by developers who are defining new selector providers or asynchronous channel providers for IO purposes.[34]	1.0 (Level 1 - "Base")
java.nio.charset	Provides components that enable mapping between sequences of sixteen bit Unicode characters and sequences of bytes representing characters. It includes capabilities for retrieving names associated with character sets, decoders to transform bytes into characters of a specific character set, and encoders to transform characters into bytes.[35]	1.0 (Level 1 - "Base")
java.nio.charset.spi	Defines "service-provider" components that would be used by developers who are creating new character sets. The service providers created would then be accessed by other developers through the java.nio.charset package.[36]	1.0 (Level 1 - "Base")
java.security	Package implements the Java security framework for providing fine-grained access control to system	1.0 (Level 1 - "Base")

[32] JAVA PLATFORM, STANDARD EDITION 7 API SPECIFICATION, http://docs.oracle.com/javase/7/docs/api/index.html (last visited January 8, 2016).

[33] PACKAGE JAVA.NIO.CHANNELS, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/nio/channels/package-summary.html (last visited January 8, 2016).

[34] PACKAGE JAVA.NIO.CHANNELS.SPI, ORACLE, http://docs.oracle.com/javase/7/docs/api/java/nio/channels/spi/package-summary.html (last visited January 8, 2016).

[35] CLASS CHARSET, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/nio/charset/Charset.html (last visited January 8, 2016).

[36] PACKAGE JAVA.NIO.CHARSET.SPI, ORACLE, http://docs.oracle.com/javase/7/docs/api/java/nio/charset/spi/package-summary.html (last visited January 8, 2016).

Copied Oracle API	API Purpose	First Android Version to Copy
	resources; for generation and storage of cryptographic public key pairs; for supporting cryptographic operations like message digests and signature generation; and for supporting signed/guarded objects and secure random number generation. At a high level, the package implements capabilities such as providing access control to prevent untrusted code from performing sensitive operations and to provide authentication.[37]	
java.security.acl*	Provides an *interface* to represent an Access Control List data structure used to guard access to resources. The list is a data structure with multiple access control entries that grant or deny access of the associated principal to underlying resources.[38]	1.0 (Level 1 - "Base")
java.security.cert	Provides components for parsing and managing certificates, certificate revocation lists (CRLs), and certification paths. Provides classes for working with identity certificates and certificate revocation lists (CRLs). It defines generic Certificate and CRL classes and X509Certificateand X509CRL classes that provide full support for standard X.509 certificates and CRLs. These classes include a simplified version of the java.security.cert package. A certificate is an object that contains the name of an entity and a public key for that entity.[39]	1.0 (Level 1 - "Base")
java.security.interfaces	Provides components to implement cryptographic algorithms. These components define methods that provide algorithm-specific information, such as key values and parameter values, about public and private keys.	1.0 (Level 1 - "Base")
java.security.spec	Provides components that specify public and private keys and encodings of those keys, a set of parameters used with digital signature algorithms, how a key can be specified.	1.0 (Level 1 - "Base")
java.sql	Provides components for accessing and processing data stored in a data source (usually a relational database). It includes a framework whereby different data source drivers can be installed dynamically to access different data sources. The	1.0 (Level 1 - "Base")

[37] PACKAGE JAVA.SECURITY, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/security/package-summary.html (last visited January 8, 2016).

[38] PAKCAGE JAVA.SECURITY.ACL, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/security/acl/Acl.html (last visited January 8, 2016).

[39] PACKAGE JAVA.SECURITY.CERT, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/security/cert/package-summary.html#package_description (last visited January 8, 2016).

Copied Oracle API	API Purpose	First Android Version to Copy
	package is mainly geared to passing SQL statements to a database, but it also provides for reading and writing data from any data source with a tabular format. The reader/writer facility, (available through the javax.sql.RowSet), can be customized to use and update data from a spread sheet, flat file, or other tabular data sources.[40]	
java.text	Provides components for handling text, dates, numbers, and messages in a manner independent of natural languages. This allows developers to write applications that are language-independent, allowing them to rely on separate, dynamically-linked resources for "localization." This allows developers to add support for additional languages at any time.[41]	1.0 (Level 1 - "Base")
java.util	Provides for the collections framework, legacy collection classes, event model, date and time facilities, internationalization, and miscellaneous utility classes (e.g. a string tokenizer, a random-number generator, and a bit array).[42] It provides the Enumeration interface for series of items (e.g. days of the week). It provides facilities for manipulating time and date information. Importantly, it provides the collections framework (library) for dealing with lists, stacks, sets and other collections.	1.0 (Level 1 - "Base")
java.util.jar	Provides components for reading and writing the JAR (Java ARchive) file format, which is an archive file whose first entry is a specially named manifest file that contains attributes and digital signatures for the file entries that follow it.	1.0 (Level 1 - "Base")
java.util.logging	Provides components related to logging. Logging is the process of writing log (A Log is record of events or information that can be tracked for future use) messages during the execution of a program to a central place. This logging allows one to report and persist so that they can later be retrieved and analyzed. This package includes: 1. Logger: The object which performs the logging in applications. 2. Level: The log levels define the severity of a	1.0 (Level 1 - "Base")

[40] PACKAGE JAVA.SQL, ORACLE https://docs.oracle.com/javase/7/docs/api/java/sql/package-summary.html (last visited January 8, 2016).

[41] PACKAGE JAVA.TEXT, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/text/package-summary.html (last visited January 8, 2016).

[42] PACKAGE JAVA.UTIL, ORACLE, https://docs.oracle.com/javase/7/docs/api/java/util/package-summary.html (last visited January 8, 2016).

Copied Oracle API	API Purpose	First Android Version to Copy
	message and which messages should be written to the log. 3. Handler: Provides a way of capturing/writing log messages to the file or on console. 4. Formatter: Specifics format for log messages to be written.	
java.util.prefs	Provides components to store and retrieve user and system preference/settings and configuration data, including user preferences and system preferences.	1.0 (Level 1 - "Base")
java.util.regex	Provides components to match or find other strings or sets of strings, using a specialized syntax held in a search pattern. The search pattern can be anything from a simple character, a fixed string or a complex expression containing special characters describing the pattern.	1.0 (Level 1 - "Base")
java.util.zip	Provides components for data compression and decompression, and for reading and writing compressed files.	1.0 (Level 1 - "Base")
javax.crypto	Provides components for cryptographic operations, including encryption, key generation and key agreement, and Message Authentication Code (MAC) generation. It supports symmetric, asymmetric, block, and stream cipher encryption, as well as secure streams and sealed objects.[43]	1.0 (Level 1 - "Base")
javax.crypto.interfaces	Provides components that support public/private key pairs and methods for two computer users to generate a shared private key with which they can then exchange information across an insecure channel. It has interfaces that specify the parameters that generate the key; define a key family and manage actual key values.	1.0 (Level 1 - "Base")
javax.crypto.spec	Provides components that specify public/private keys and encodings of those keys, as well as how a key can be specified. A key may be specified in an algorithm-specific way, or in an algorithm-independent encoding format.	1.0 (Level 1 - "Base")
javax.net	Provides components for networking applications, including for factories for creating sockets. Socket factories allow developers to encapsulate socket creation and configuration behavior.[44] This Factory design provides polymorphism, makes it more	1.0 (Level 1 - "Base")

[43] PACKAGE JAVA.CRYPTO, ORACLE, https://docs.oracle.com/javase/7/docs/api/javax/crypto/package-summary.html (last visited January 8, 2016).

[44] PACKAGE JAVAX.NET, ORACLE, http://docs.oracle.com/javase/7/docs/api/javax/net/package-summary.html (last visited January 8, 2016).

Copied Oracle API	API Purpose	First Android Version to Copy
	flexible, so different kinds of sockets can be used by the same application code just by passing it different kinds of factories.	
javax.net.ssl	Provides components that support secure infrastructure for networking applications. Using these classes and interfaces one can communicate reliably into the network and can optionally encrypt the data and/or authenticate the communicating peers.	1.0 (Level 1 - "Base")
javax.security.auth	Provides components for authentication and authorization. It supports various types of authentication modules. The authorization component allows specification of access controls.	1.0 (Level 1 - "Base")
javax.security.auth.callback	Provides components for services to interact with applications in order to retrieve information (authentication data including usernames or passwords, for example) or to display information (error and warning messages, for example).	1.0 (Level 1 - "Base")
javax.security.auth.login	Provides components for configuration related to login and authentication.	1.0 (Level 1 - "Base")
javax.security.auth.x500	Provides components that should be used to store X.500 principals and their credentials. X.500 is a series of protocols for computer networking covering electronic directory services.	1.0 (Level 1 - "Base")
javax.security.cert	Provides components for working with identity certificates and certificate revocation lists (CRLs). A certificate is an object that contains the name of an entity and a public key for that entity. This was an extension for java.security.cert but was added to core J2SE 1.4	1.0 (Level 1 - "Base")
javax.sql	This package supplements the java.sql to provide capabilities for server side access and processing of data from data sources. Similar to the java.sql package it provides ways to make connections to data sources, including relational databases. It includes greater flexibility for making changes to a data source's properties and additional capability for connection and statement "pooling" and distributed database transactions.[45]	1.0 (Level 1 - "Base")

[45] PACKAGE JAVAX.SQL, ORACLE, http://docs.oracle.com/javase/7/docs/api/javax/sql/package-summary.html (last visited January 8, 2016).

XI. APPENDIX J: ANDROID API LEVELS AND VERSIONS[46]

API Level	Version
23	Marshmallow
22 21	Lollipop
20 19	KitKat
18 17 16	Jelly Bean
15 14	Ice Cream Sandwich
13 12 11	Honeycomb
10 9	Gingerbread
8	Froyo
7 6	Eclair
5 4	Donut
3	Cupcake
2 1	Base

[46] <USES-SDK>, ANDROID, http://developer.android.com/guide/topics/manifest/uses-sdk-element.html (last visited January 8, 2016).